



VYSOKÁ ŠKOLA BÁŇSKÁ – TECHNICKÁ UNIVERZITA OSTRAVA

EKONOMICKÁ FAKULTA

KATEDRA APLIKOVANÉ INFORMATIKY

Racionalizace informačního systému sportovního klubu

(Rationalization of a Sport Club's Information System)

Student:

Pavel Rakús

Vedoucí bakalářské práce

Ing. Vítězslav Novák, PhD.

2012

## Zadání bakalářské práce

Student: **Pavel Rakús**  
Studijní program: B6209 Systémové inženýrství a informatika  
Studijní obor: 6209R001 Aplikovaná informatika  
Téma: Racionalizace informačního systému sportovního klubu  
Rationalization of a Sport Club's Information System

Zásady pro vypracování:

1. Úvod
2. Teoretická východiska práce
3. Představení a rozbor současné verze
4. Návrh řešení nové verze (PHP, MySQL) a jeho realizace
5. Zhodnocení řešení a porovnání
6. Závěr

Seznam použité literatury

Seznam zkratk

Prohlášení o využití výsledků bakalářské práce

Seznam příloh

Přílohy

Seznam doporučené odborné literatury:

BORONCZYK, T., E. NARAMORE, J. GERNER, Y. LE SCOUARNEC, J. STOLZ a M. K. GLASS.  
*PHP 6, MySQL, Apache: Vytváříme webové aplikace.* Přeložil Bogdan KISZKA. Brno: Computer Press, 2011. ISBN: 978-80-251-2767-4.

CEDERHOLM, Dan. *Webdesign s webovými standardy.* Přeložil Jaroslav BLAŽEK. Brno: Zoner Press, 2004. ISBN: 80-86815-15-3.

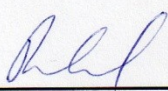
DOMES, Martin. *333 tipů a triků pro CSS.* Brno: Computer Press, 2009. ISBN: 978-80-251-2360-7.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

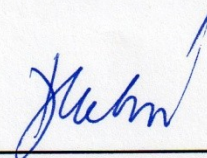
Vedoucí bakalářské práce: **Ing. Vítězslav Novák, Ph.D.**

Datum zadání: 25.11.2011

Datum odevzdání: 11.05.2012

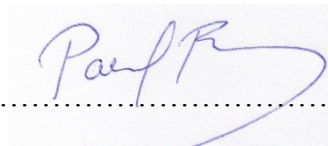
  
Ing. Petr Rozehnal, Ph.D.  
vedoucí katedry



  
prof. Dr. Ing. Dana Dluhošová  
děkanka fakulty

Prohlašuji, že jsem celou práci, včetně všech příloh, vypracoval samostatně.

Přílohy č. 1 a 2, dané mi k dispozici, jsem samostatně doplnil.

A handwritten signature in blue ink, appearing to read 'Pavel Rakús', is written over a horizontal dotted line.

Pavel Rakús

V Ostravě dne 11. 5. 2012

## Obsah

1	Úvod.....	6
2	Teoretická východiska práce.....	7
2.1	World Wide Web (WWW).....	7
2.1.1	Vznik WWW.....	7
2.2	Tři základní pilíře webových stránek .....	7
2.3	HTML.....	8
2.3.1	Historie a vývoj jazyka HTML .....	8
2.3.2	Pravidla syntaxe HTML .....	9
2.3.3	Struktura stránky v jazyce HTML.....	10
2.3.4	Formuláře v HTML.....	10
2.4	Kaskádové styly.....	11
2.4.1	Historie kaskádových stylů .....	12
2.4.2	Použití kaskádových stylů.....	12
2.4.3	Přiřazení CSS elementu HTML .....	14
2.5	PHP.....	14
2.5.1	Popis a historie jazyka PHP .....	14
2.5.2	Server vs. klient.....	15
2.5.3	Interpretovaný vs. kompilovaný kód.....	15
2.5.4	Použití jazyka PHP.....	16
2.5.5	Struktura jazyka PHP .....	16
2.5.6	Vkládání PHP do HTML kódu.....	17
2.5.7	Komentáře v jazyce PHP.....	18
2.5.8	Konstanty a proměnné v jazyce PHP .....	19
2.5.9	Funkce v jazyce PHP.....	20
2.5.10	Podmínky v jazyce PHP .....	20
2.5.11	Cykly v jazyce PHP.....	21
2.5.12	Soubory cookies v PHP .....	21
2.5.13	Sessions v PHP.....	22
2.5.14	Práce s databází v PHP.....	23
2.6	Databáze .....	23
2.6.1	Klíče .....	24
2.6.2	Relace a referenční integrita.....	24

2.6.3	Datové typy v mySQL.....	24
2.6.4	Práce se strukturou a daty v databázích.....	25
3	Představení a rozbor současné verze.....	28
3.1	Představení šachového klubu.....	28
3.2	Historie stránek šachového klubu.....	28
3.3	Představení současné verze.....	32
3.3.1	Řešení rozvržení obsahu na stránce.....	33
3.3.2	Formátování stránek.....	34
3.3.3	Administrační rozhraní.....	34
3.3.4	Správa článků, akcí a archivu.....	35
3.3.5	Správa návštěvní knihy a družstev.....	35
3.3.6	Seznam členů.....	36
3.3.7	Galerie fotografií.....	36
3.3.8	Návštěvní kniha.....	36
3.3.9	Ostatní stránky.....	37
3.3.10	Návrh databáze.....	37
3.4	Rozbor současné verze.....	37
3.4.1	Problémy tabulkového layoutu.....	38
3.4.2	Problémy se systémem uživatelských účtů.....	38
3.4.3	Chyby v galerii fotografií.....	38
3.4.4	Nedostatky v kontrole zadaných údajů.....	39
3.4.5	Problémy při využívání databáze.....	39
3.4.6	Nedostatky u seznamů.....	39
3.4.7	Omezená míra automatizace.....	39
3.4.8	Nedostatky v kódování.....	39
3.4.9	Grafické zpracování.....	40
4	Návrh řešení nové verze (PHP, MySQL) a jeho realizace.....	41
4.1	Produkční plán webu.....	41
4.1.1	Název webu a adresa.....	41
4.1.2	Záměr a cíle nového webu.....	41
4.1.3	Cílové publikum a návštěvnost.....	42
4.1.4	Tým, konkurence, rozpočet.....	43
4.2	Návrh grafického uspořádání stránek.....	43

4.3	Kódování v HTML a validita .....	46
4.4	Návrh databáze MySQL .....	46
4.4.1	Návrh struktury a datových typů tabulek .....	46
4.4.2	Vazby mezi tabulkami.....	47
4.4.3	Zálohování databáze.....	48
4.5	Programové řešení v PHP .....	49
4.5.1	Kombinace HTML a PHP .....	49
4.5.2	Registrace a přihlašování uživatelů do systému.....	49
4.5.3	Administrační rozhraní.....	51
4.5.4	Redakční systém.....	52
4.5.5	Práce s databází .....	53
5	Zhodnocení řešení a porovnání .....	54
5.1	Zhodnocení řešení.....	54
5.2	Porovnání se starší verzí .....	54
5.2.1	Grafické porovnání.....	54
5.2.2	Porovnání funkčnosti.....	55
6	Závěr .....	56
	Seznam použité literatury.....	57
	Seznam zkratk.....	58
	Prohlášení o využití výsledků diplomové (bakalářské) práce	
	Seznam příloh	

# 1 Úvod

Internet a jeho nejrozšířenější a nejvyužívanější služba WWW od svého zavedení v 90. letech minulého století urazily obrovský kus cesty – z univerzit a vědeckých pracovišť se postupně přemístily do soukromého sektoru, kde si následně svými inovátorskými myšlenkami a možnostmi získaly přízeň mnoha stovek milionů uživatelů.

Pro každého moderního člověka prahnoucího po informacích a spojení s okolním světem je dnes internet běžnou součástí jeho života. Komunikace a vyhledávání potřebných informací v této síti tvoří pracovní náplň velké většiny lidí, kteří se v zaměstnání bez počítače neobejdou. Vždyť kdo by v dnešní době neznal např. elektronickou poštu, tzv. e-mail, přes kterou proudí obrovské množství zpráv a jiných dat, nebo různé sociální sítě, zpravodajské servery, sdílení souborů a podobné služby. Velkou a důležitou roli v rámci internetu hrají samozřejmě také prezentace jednotlivých subjektů, ať už to jsou státní instituce, komerční společnosti nebo různé soukromé fyzické i právnické osoby.

Šachový klub TJ MSA Dolní Benešov je sportovní oddíl, spadající pod tělovýchovnou jednotu místní akciové společnosti, která se zabývá výrobou průmyslových armatur a čerpadel. Klub byl založen roku 1947 a příslušnost k němu v současnosti přiznává každým rokem kolem 50 aktivních hráčů. I přesto, že město Dolní Benešov má pouhé čtyři a půl tisíce obyvatel, patří klub k absolutní špičce v okrese Opava a je silným konkurentem i mnohem větším šachovým oddílům v Moravskoslezském kraji.

Potřeba vlastní prezentace na internetu byla poprvé vedením klubu diskutována v roce 2005, kdy byla také spuštěna první verze internetových stránek klubu. Aktualizace stránek byla nutností – první přišla hned roku 2005, v roce 2008 pak přišla zcela nová, dynamická verze stránek, která s mírnými úpravami funguje dodnes.

Cílem této bakalářské práce je zmapování a rozbor současné verze, analýza chyb této verze a návrh řešení aktualizované verze stránek a jeho následná realizace.

Bakalářská práce je rozdělena do čtyř hlavních částí. V první části jsou popsána teoretická východiska a použité techniky tvorby webového portálu. Ve druhé části je analyzována současná verze, je poukázáno na její nedostatky. Ve třetí části je navrhováno řešení nové verze, které je pak v poslední čtvrté části porovnáváno s původní verzí. Na základě tohoto porovnání pak probíhá zhodnocení, zde bylo dosaženo toho, co si tato práce kladla za cíl.



## 2 Teoretická východiska práce

V této části práce jsou popsány teoretické předpoklady pro tvorbu dynamických webových stránek v jazyce PHP za použití relačních databází. Jsou zde popsány základní vlastnosti a parametry daných technologií, jejich historie i příklady použití.

### 2.1 World Wide Web (WWW)

World Wide Web, WWW, někdy také W3, nebo „celosvětová pavučina“ je dnes nejrozšířenější služba či určitá nadstavba, náplň internetu, *grafický internet*. Web využívá technické struktury internetu (počítače v celosvětové síti, možnost komunikace kohokoli s kýmkoli v této síti) a zprostředkovává informace uložené na těchto počítačích v atraktivní grafické podobě. [5]

#### 2.1.1 Vznik WWW

První kapitola webu byla napsána roku 1989 ve švýcarském CERNu. V tomto centru definoval Tim Bernes-Lee hypertextový systém, první intranet na světě. O rok později napsal Bernes-Lee první program pro tvorbu webových stránek a pro systém, běžící na jednom počítači návrh název „World Wide Web“. Dále pak už jsou roky nepřetržitého růstu, dosahujícího místy rozměrů epidemie. V roce 1992 existuje na světě okolo padesáti webových serverů, vznikají první grafické prohlížeče, Midas a Viola. O rok později jsou však přetrumfnuty skvělým dílkem Marca Andreese a Erica Biny, autorů prohlížeče Mosaic. Další rok vznikají zárodky firmy Netscape, vznikají první dedikované programy, je ustanoveno konsorcium W3C; v roce 1995 je na světě již 100 000 webových serverů a web se stává hlavním proudem Internetu. [5]

### 2.2 Tři základní pilíře webových stránek

Celý internet a také služba WWW jsou založeny na používání norem (standardů), které popisují způsob komunikace v internetu, formáty používaných datových struktur atd. Tyto normy přitom nejsou vázány na určité specifické vlastnosti daného operačního systému, takže je lze ve většině případů implementovat na všech používaných platformách. A právě nezávislost na platformě je jednou z nejdůležitějších vlastností služby WWW. [6] A jaké jsou hlavní pilíře WWW?

*HTML (HyperText Markup Language)*. První definice tohoto značkovacího jazyka byla už u zrodu samotného projektu WWW v roce 1991 a jejím autorem byl stejně jako u

celého konceptu WWW Tim Bernes-Lee. Přes různé úpravy, které souvisely také s vývojem webových prohlížečů a jejich zpětnou vazbu na vývoj jazyka, se jazyk HTML propracoval do současně používané verze HTML 4.01. [6]

*URL (Uniform Resource Locator)*. WWW stránky vytvářejí hypertextový informační systém. Struktura hypertextu je tvořena odkazy mezi stránkami. Tyto odkazy se zapisují také ve standardizované podobě a to pomocí URL. [7]

*HTTP (HyperText Transfer Protocol)*. Protokol HTTP se používá při komunikaci klienta (webový prohlížeč) a serveru. Touto komunikací se rozumí především výměna hypertextových dokumentů. Protokol HTTP je v současné době nejpoužívanějším protokolem internetu, ve většině případů používá port TCP/80. Celý proces komunikace probíhá takovým stylem, že:

- klient naváže se serverem komunikace a zašle mu svůj požadavek,
- server pošle klientovi odpověď na jeho požadavek.

Protokol HTTP existuje ve verzích 0.9, 1.0 a 1.1, které se v detailech liší. Pro zabezpečenou komunikaci vznikl protokol HTTPS, který využívá protokol HTTP a šifrování SSL příp. TLS. [6]

## **2.3 HTML**

HTML, HyperText Markup Language, je název jazyka, který je použit k popisu webových stránek. Jeho podstatné dvě složky jsou *hypertext*, tj. odkazy na externí elementy, a *markup*, tj. využívání značek (tagů) pro formátování dokumentu. [5]

### **2.3.1 Historie a vývoj jazyka HTML**

První definice jazyka HTML vytvořil Tim Bernes-Lee již v roce 1991 jako součást projektu WWW, jehož prvotním cílem bylo umožnit vědcům, zabývajícím se fyzikou vysokých energií, komunikaci a sdílení výsledků výzkumů po celém světě. Celý tento projekt vznikl ve švýcarském CERNu, evropském centru jaderného vývoje, nacházejícím se nedaleko Ženevy. Prvotní verze HTML byla známa pod označením HTML 0.9 a umožňovala rozčlenit text do několika logických úrovní, použít několik druhů zvýraznění textu a zařadit do textu obrázky a také odkazy. [6]

První verze WWW software byla napsána pro operační systém NextStep a obsahovala jak prohlížeč, tak integrovaný editor WWW stránek. Od tohoto řešení se později upustilo. [6]

Časem požadavky uživatelů na WWW vzrůstaly, a tak producenti různých webových prohlížečů obohacovali jazyk HTML o některé nové, vlastní prvky. Z důvodu zachování kompatibility mezi jednotlivými modifikacemi HTML vytvořil Bernes-Lee návrh standardu HTML 2.0, který zahrnoval všechny v té době běžně používané prvky HTML. Na jaře 1995 vznikl návrh standardu HTML 3.0, přinášející velké rozšíření. Některé prvky HTML 3.0 podporovaly novější verze prohlížečů Netscape a Mosaic, kompletní podporu nabízel pouze prohlížeč Arena, dostupný pod operačními systémy Unix. Na počátku roku 1996 bylo jasné, že HTML 3.0 bylo tak mohutným skokem vpřed, že se nenašel nikdo, kdo by dokázal implementovat jeho kompletní podporu do svého prohlížeče. Vývoj standardu WWW byl v té době již řízen konsorciem W3C. Jeho členové se shodli, že bude vytvořena nová verze, vycházející ze standardu HTML 2.0 – HTML 3.2. Tato nově vytvořená verze však zdaleka neobsahovala všechny prvky z HTML 3.0. Stalo se v podstatě to stejné, co při vzniku verze 2.0 – jazyk byl sjednocen na průniku možností tehdejších nejrozšířenějších prohlížečů. [6]

V létě roku 1997 vstoupila do povědomí verze HTML 4.0, na které se velcí vývojáři společně s konsorciem W3 sjednotili a v prosinci téhož roku pak byl tento standard schválen a přijat. [5]

24. prosince 1999 byla konsorciem W3 doporučena verze HTML 4.01, která se používá jako standard pro dokumenty HTML i dnes. V současné době se pracuje na zavedení revoluční verze HTML5 (v návrhu od roku 2008).

### **2.3.2 Pravidla syntaxe HTML**

HTML má samozřejmě danou svou syntaxi. Její pozoruhodnost a unikátnost spočívá v tom, že je sice přesně nadefinována různými standardy, ale zároveň je velice flexibilní: i pokud je v dokumentu mnoho chyb lze odhalit a detekovat správné části kódu. Značné množství webových stránek nespĺňuje přesná pravidla HTML, přesto to nebrání jejich zobrazování a prohlížení. [5]

Jazyk HTML má textový (ASCII) formát a přímo v tomto formátu je používán – není kompilován do binární ani žádné jiné podoby. Jakmile je vytvořena stránka HTML v textovém editoru, je ve finálním tvaru, který přímo čte a interpretuje webový prohlížeč.

Soubory, které jsou vždy v binárním formátu, např. obrázky, zvukové soubory či jiné objekty, nejsou začleněny do HTML souborů – v nich jsou uvedeny pouze odkazy na tyto soubory. [5]

### 2.3.3 Struktura stránky v jazyce HTML

Jak již bylo zmíněno, základními dvěma prvky webových dokumentů v HTML jsou *hypertext* a *markup*. Markup neboli značkování, má na stránce svá pravidla, provádí se pomocí tzv. tagů (značek). Ty mohou být párové nebo nepárové. Existuje jich nepřeberné množství a na tomto místě není možno je všechny do detailu popisovat. Základní struktura HTML dokumentu je zobrazena na následujícím příkladě:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
..... označení typu dokumentu a verze HTML
<html> ..... začátek textu ve formátu HTML
<head> ..... začátek hlavičky
<meta http-equiv="content-type" content="text/html; charset=utf-8">
..... meta-informace, data o dokumentu (např. znaková sada)
<title>Dokument HTML</title> ..... název dokumentu (zobrazuje se v liště)
</head> ..... konec hlavičky
<body> ..... začátek těla stránky
..... OBSAH STRÁNKY .....
</body> ..... konec těla stránky
</html> ..... konec textu ve formátu HTML
```

V obsahu stránky jsou umístěny zmiňované párové a nepárové tagy, např. nadpisy, odstavce s textem, hypertextové odkazy na jiné dokumenty, tabulky s daty, obrázky atd. [5]

### 2.3.4 Formuláře v HTML

Formuláře jsou velice důležitým prvkem HTML. Umožňují uživateli zadávat data, které lze následně zpracovat pomocí skriptovacího jazyka (např. PHP). Formulář je v HTML uzavřen v elementu `<form>`. Element `<form>` je nevizuální, nijak se neprojeví na vzhledu stránky. Má několik atributů, z nichž nejdůležitější jsou dva: *action* a *method*. Atribut *action* určuje URL aplikace (skriptu), pomocí které je odeslaný výsledek formuláře zpracován. Druhým důležitým atributem je atribut *method*, kterým je určen způsob, jakým jsou data z formuláře skriptu zaslána. Možnými hodnotami tohoto atributu jsou *get* a *post*. Metoda *get* předává data přímo v URL, zatímco metoda *post* je zašle v hlavičce HTTP požadavku. Výhodou a zároveň nevýhodou metody *get* je, že všechna předaná data jsou v URL stránky

vidět. Výhoda spočívá v tom, že takovéto URL lze uložit a opět znovu obnovit později s již uloženými hodnotami. Lze toho využít např. u vyhledávacích služeb. Nevýhodu představuje zobrazení citlivých údajů v URL stránky, např. při přihlášení do zabezpečené zóny webu. Pro takové případy se používá metoda *post*. Výhodou metody *post* je i možnost posílat prakticky libovolně dlouhá data, včetně textových i binárních souborů. [2]

Nejčastěji používaným formulářovým prvkem je element `<input>`, který je zároveň nejvíce variabilním prvkem pro použití ve formulářích. Reprezentuje několik aktivních prvků – který z nich se nakonec zobrazí na stránce je určeno atributem *type*. Možnosti, které se používají a jsou využity i v projektu jsou:

- *text* – běžné, jednořádkové textové vstupní pole,
- *password* – textové pole pro zadávání hesla (nezobrazuje znaky),
- *checkbox* – zaškrťovací políčko,
- *radio* – přepínač,
- *hidden* – skryté pole (slouží k předávání proměnných mezi stránkami),
- *submit* – tlačítko pro odeslání formuláře,
- *reset* – tlačítko pro vymazání hodnot formuláře,
- *image* – tlačítko s obrázkem,
- *file* – vstup pro soubor,
- *button* – tlačítko bez obecně definované funkce.

Dalšími atributy elementu `<input>` jsou především atributy *name* a *value*. *Name* určuje jméno prvku, které musí být v rámci dokumentu jednoznačné. Zároveň je pod tímto jménem hodnota přenesena zpracovávacímu skriptu a je dostupná v PHP. Atribut *value* může určovat hodnotu, kterou bude mít prvek ještě před vlastním zobrazením na stránce. [2]

## 2.4 Kaskádové styly

CSS je zkratkou *Cascading Style Sheet*, česky tedy kaskádové styly. Kaskádové styly jsou jazykem, který je určen pro popis vzhledu elementů jazyků HTML, XHTML a XML. [4]

Cílem vzniku CSS bylo oddělení vzhledu webové stránky od jejího významu. Úkolem kaskádových stylů pak je popisovat vzhled prvků stránky – na rozdíl od značkovacích jazyků (HTML, XHTML, XML), jejichž úkolem je dodávat obsahu význam. [4]

#### 2.4.1 Historie kaskádových stylů

Jazyk CSS navrhla společnost W3C (World Wide Web Consortium). Autorem prvního návrhu byl Håkon Wium Lie. V současné době existují dvě hotové verze – CSS1 a CSS2. V roce 2011 byla dokončena revizní verze CSS 2.1, která je přijímána jako standard jazyka. V současné době se pracuje na další verzi CSS3. [4]

#### 2.4.2 Použití kaskádových stylů

Pro použití kaskádových stylů v dokumentech (x)HTML existují čtyři způsoby aplikace. Každá z nich má své výhody i nevýhody, v závislosti na situaci se můžeme rozhodnout pro použití každé z těchto možností. [3]

##### Metoda A: Prvek `<style>`

Tato metoda je také známá pod označením „vložený seznam stylů“ (*embedded style sheet*). Umožňuje zapsat všechny deklarace CSS přímo do (X)HTML dokumentu. Prvek `<style>` je umístěn uvnitř oddílu `<head>` a může obsahovat libovolný počet pravidel CSS.

```
<head>
<style type="text/css">
<![CDATA [
... zde jsou deklarace CSS ...
] ]>
</style>
</head>
```

Atribut `type` s hodnotou `text/css` zajišťuje, že prohlížeč rozumí tomu, jaký typ jazyka stylu překládá a jaký je vyžadován. Také se používá syntaxe s komentářem `CDATA` doporučená W3C, která umožňuje schovat pravidla stylu před staršími prohlížeči, které nejsou schopny jim porozumět. [3]

Jako každá metoda má toto použití CSS své výhody i nevýhody. Za silnou stránku můžeme považovat to, že v začátcích tvorby a během testování CSS v dokumentu HTML je

šikovné psát všechna pravidla podle metody A. Pracujeme pak pouze s jedním dokumentem a díky tomu můžeme měnit jak samotný kód, tak i styly. Po dokončení testování pak můžeme CSS zveřejnit pomocí jiné metody. Naopak k nevýhodám metody A patří skutečnost, že některé starší prohlížeče mohou mít problémy při zpracování pokročilejších pravidel pro grafické rozvržení a určení polohy. Další nepříznivou vlastností vložených seznamů stylů je nutnost stáhnout je při každém načítání stránky – jsou totiž součástí kódu dané stránky. [3]

### **Metoda B: Externí seznamy stylů**

```
<head>
<link rel="stylesheet" type="text/css" href="style.css">
</head>
```

Pomocí metody B je demonstrováno spojení s externími seznamy stylů, ve kterých se v samostatném souboru nacházejí všechny deklarace CSS. Odkaz na ně je pomocí prvku `<link>` umístěn v hlavičce dokumentu HTML. Atribut `href` pak ukazuje umístění souboru. Hodnota může být relativní cestou (viz v kódu výše) nebo absolutní cestou, tj. plné umístění `http:// dokumentu`. [3]

Mít všechna pravidla CSS v souboru odděleném od vlastního kódu představuje jednu nespornou výhodu – jakékoliv změny stylu celého webového serveru lze učinit v tomto jediném souboru; není potřeba mít na každé stránce tytéž deklarace CSS, jak by tomu muselo být v případě metody A. Toto je samozřejmě kritický aspekt hlavně u rozsáhlých webových serverů, kde mohou pokyny ke stylu sdílet stovky a tisíce stránek. Další výhodou odkazování stylů do externího seznamu stylů je to, že soubor se styly je prohlížečem stažen pouze jednou a uložen. Tím se šetří čas při opakovaných návštěvách stránek nebo při prohlížení stránek, které odkazují na totožný seznam stylů. Nevýhodou může být opět implementace CSS stylů některými staršími prohlížeči. [3]

### **Metoda C: Styly uvnitř kódu**

```
<h1 style="font-family: Arial, Verdana; color: orange;">Toto je nadpis</h1>
```

Existuje také třetí metoda použití CSS, o které je potřeba se zmínit – styly uvnitř kódu. Metoda C ukazuje, že atribut `<style>` lze přiřadit ke každému prvku, čímž je pravidlům CSS umožněno, aby byly aplikovány na jakoukoliv úroveň prvků. Protože styly v kódu jsou

kaskádě na nejnižší možné úrovni, zruší platnost všech elementů, které jsou deklarovány v externích seznámech stylů nebo pravidel obsažených v prvku `<style>` v hlavičce dokumentu. I u této metody se vyskytují výhody a nevýhody. Nevýhodou je bezesporu to, že při používání stylů přímo uvnitř kódu fakticky není oddělen obsah od vzhledu. Modifikace vzhledu pak znamená nutnou editaci samotného kódu. Oddělení CSS do zvláštního souboru údržbu usnadňuje. Metoda C je proto vhodná pro dočasná řešení nebo pro případy, kdy je přístup k externímu souboru části `<head>` dokumentu nemožný. [3]

### 2.4.3 Přiřazení CSS elementu HTML

K pojmenování elementů v dokumentu HTML k formátování pomocí stylů lze v CSS použít buď třídu, nebo identifikátor. Oba prvky je možné použít v identických situacích a výsledek bude stejný. Přesto zde existuje určitý rozdíl:

- Třída se používá k volnému pojmenování elementů na stránce a třídu lze v jednom dokumentu HTML použít vícekrát.
- Identifikátor slouží k jednoznačnému pojmenování a v jednom dokumentu HTML smí být použit pouze jednou.

Třída se obvykle používá k obecnému pojmenování elementů stránky, identifikátor naopak pro jednoznačné pojmenování bloků stránky – záhlaví, zápatí, obsah apod.

## 2.5 PHP

PHP (zkratka Personal Home Page) je hypertextový preprocesor, který na serveru interpretuje stránky HTML s vlastními příkazy před jejich odesláním ke klientovi (tradičně webový prohlížeč). [2] Jeho všestrannost a relativně nízká křivka osvojení z něj činí jeden z nejpopulárnějších současně používaných skriptovacích jazyků. [1]

### 2.5.1 Popis a historie jazyka PHP

Počátky PHP se datují od roku 1994. Tehdy byl programátorem Rasmusem Lerdorfem napsán skript pro počítání přístupů k jeho stránkám, který byl založen na jazyku Perl (později byl přepsán do jazyka C). Sada těchto skriptů byla posléze vydána pod názvem *Personal Home Page Tools* nebo častěji také *PHP Tools*. V roce 1995 se poté systém PHP propojil s dalším systémem nazvaným Form Interpreter (zkráceně FI), čímž dal vzniknout systému PHP/FI 2.0, který si získal celosvětovou oblibu. [8]



Roku 1998 je vydána verze PHP 3.0, běžící také pod systémy Windows. V zimě 1998 začíná práce na nové verzi, která vrcholí v polovině roku 1999 vydáním verze PHP 4.0, přinášející např. podporu HTTP Sessions. Verze 5.0 byla vydána v červenci roku 2004, aktuální verzí je 5.3.10. [8]

PHP je technologie běžící na straně serveru, na rozdíl např. od JavaScriptu, který využívá stanice klienta. V praxi běžný skript PHP kódu obsahuje kombinace standardního HTML kódu a kusy programového kódu. Když pak webový server obdrží požadavek na zpracování tohoto skriptu, provede části programového kódu v PHP a zkombinuje je s původním kódem v HTML, načež tento výsledek odešle zpět webovému prohlížeči. V prohlížeči je pak již zobrazena výsledná verze vygenerované stránky, která může obsahovat např. požadované informace z databáze, které byly získány pomocí skriptu PHP. [9]

### **2.5.2 Server vs. klient**

Je nesmyslné řešit, zda je lepší programový kód provádět na straně serveru či na straně klienta, protože každá z těchto možností má svoje pro i proti. Provádění kódu na serveru přináší především výhody, ke kterým patří:

- snadná interakce s dalšími aplikacemi na serveru,
- snadnější zabezpečení celého systému,
- nenáročnost na hardware a software klienta (výstupem je čisté HTML),
- menší objem přenesených dat,
- výrazně vyšší ochrana zdrojových textů programu.

Naopak provádění kódu u klienta má také několik výhod. Jednou z hlavních je menší zátěž serveru. Server pouze dostane požadavek a odešle stránku, zatímco v případě serverových technologií musí být ještě spuštěn interpret jazyka, musí mu být předána stránka a následně klientovi zaslán výsledek. Další výhodou je možnost interakce s objekty na webové stránce. Klientské technologie vznikají především z tohoto důvodu. [2]

### **2.5.3 Interpretovaný vs. kompilovaný kód**

Jazyk PHP je interpretovaný, což znamená, že až do okamžiku svého spuštění je kód uchováván ve zdrojovém tvaru. Interpret jazyka tento kód vezme a překládá jej do strojového kódu pro počítač, na kterém PHP běží. To je samozřejmě pomalejší než u kompilovaného

kódu, který je k dispozici přímo ve spustitelném tvaru. Na internetu je možno setkat se s oběma přístupy. [2]

Výhodu kompilovaných jazyků je jejich vyšší rychlost, neboť pro své spuštění nevyžadují „tlumočníka“. Na druhou stranu nejsou tyto programy přenositelné mezi různými operačními systémy. Výhodou interpretovaných jazyků je např. možnost snažších úprav – stačí provést úpravu ve zdrojovém souboru, který může být vždy k dispozici. V prvním případě je po provedení změny potřeba ještě program znovu přeložit, aby byla změna skutečně provedena. [2]

#### **2.5.4 Použití jazyka PHP**

A co vše se dá v PHP napsat? V podstatě všechno, co nějakým způsobem souvisí s dynamickými webovými stránkami a databázemi. Nejčastěji se tedy jazyka PHP využívá k psaní aplikací, jako jsou:

- internetové obchody,
- redakční systémy,
- dynamické osobní / firemní stránky,
- weboví poštovní či databázoví klienti,
- různé vyhledávače a katalogy webových stránek,
- firemní informační systémy,
- diskusní fóra,
- drobnosti jako počítadla, ankety a podobně. [8]

#### **2.5.5 Struktura jazyka PHP**

Stejně jako stránky HTML lze také programy v jazyce PHP psát i v jednoduchém textovém editoru, jako je např. Poznámkový blok nebo PSPad. Soubory se liší v příponě, ty s obsahem kódu PHP musí mít příponu *\*.php*. Tato přípona indikuje, že soubor musí být předán skriptovacímu stroji PHP a až výsledek této činnosti lze odeslat zpět do webového prohlížeče. [1]

Kód v jazyce PHP se liší od jazyka JavaScript nebo jiných podobných skriptovacích jazyků, které bývají interpretovány v dokumentu HTML uvnitř značkovacího kódu jazyka HTML. PHP neumožňuje pouze dynamické generování HTML stránek, kód v tomto jazyce je

navíc pro návštěvníky webu zcela neviditelný. Návštěvník vidí pouze výstup, což je čistý kód HTML. V tomto ohledu poskytuje PHP o něco větší bezpečí, spočívající ve skrytí programové logiky webové aplikace. [1]

PHP je strukturovaný jazyk, což v zásadě znamená, že spolu související příkazy (tím se rozumí příkazy prováděné za sebou) je možné sdružovat do tzv. bloků, které tvoří celkovou strukturu programu. Bloky příkazů jsou ohraničeny složenými závorkami a používají se např. při vyhodnocení podmíněných příkazů (pokud podmínka platí, proved' blok příkazů, pokud neplatí, proved' jiný blok příkazů), pro ohraničení funkce, těla cyklu atd. Bloky je možno do sebe také vnořovat.

```
<?php
function funkce() {
    if (podmínka) {
        příkaz1;
        příkaz2;
        příkaz3;
    }
}
?>
```

### 2.5.6 Vkládání PHP do HTML kódu

PHP umožňuje vkládat svůj programový kód přímo do zdrojového kódu dokumentů, napsaných v jazyce HTML. Z toho logicky vyplývá, že musí existovat způsob, jak oddělit značkování HTML od kódu PHP, aby interpret jazyka PHP poznal, která část je určena pro něj a která má být ponechána beze změny a zaslána klientskému programu (tím se rozumí v našem případě prohlížeč, ale nemusí to být vždy pravda). [2]

Možností, jak oba jazyky (HTML a PHP) na stránce oddělit, je několik. Standardní způsob používá řetězec <?php jako počáteční oddělovač kódu a ?> jako ukončovací.

```
... kód HTML ...

<?php
... kód PHP ...

?>

... kód HTML ...
```

Obvykle je možné používat také zkrácenou formu zápisu, kde počáteční oddělovačem je pouze dvojznak `<?`, tento způsob však není kompatibilní s jazykem XML a není jej tedy dost dobře možné použít pro značkování PHP uvnitř XML dokumentů.

```
... kód HTML ...  
<? ... kód PHP ... ?>  
... kód HTML ...
```

Třetí způsob, jak vložit PHP do stránky, představuje element `SCRIPT` jazyka HTML.

```
<script language="php">  
...  
</script>
```

Výhodou tohoto způsobu zápisu je, že s ním žádná zpracující aplikace nemá problémy. Naopak zjevnou nevýhodou může být zbytečně dlouhý zápis oproti oběma předchozím způsobům. To se projeví zvláště v případě, kdy není celá stránka tvořena pouze příkazy PHP, ale opakovaně se vstupuje z HTML kódu do PHP a zpět. [2]

V projektu je použito pouze prvního způsobu vkládání, a to především z důvodů lepší přehlednosti a zkrácení délky zápisu.

### 2.5.7 Komentáře v jazyce PHP

Poměrně často se vyskytuje potřeba udělat v kódu vysvětlující poznámku k danému příkazu nebo bloku příkazů, tzv. komentář. V PHP je možné používat tři druhy komentářů. Standardní komentář pochází z jazyka C, ve kterém je jako komentář bráno vše mezi znaky `/*` a `*/`. Tento komentář je víceřádkový, to znamená, že může být rozložen do několika řádků. [2]

Další možností je použití komentářů používaných v C++ a dalších jazycích – v tomto případě se jako uvozovací řetězec pro komentář používá dvojice lomítek `(//)` a za komentář jsou pak považovány všechny znaky až do konce řádku. [2]

Třetí způsob, v projektu ovšem nepoužitý, vychází ze shellových skriptů pro operační systém Linux – jako uvozovací znak se používá # a komentář je opět vše do konce řádku. [2]

Komentáře lze do sebe také vnořovat, týká se to především komentářů /\* a \*/. Uvnitř komentáře /\* a \*/ lze bez problémů použít ostatní řádkové druhy komentářů, nikoliv však ten samý – v takovém případě je zpracování komentáře předčasně ukončeno. [2]

### **2.5.8 Konstanty a proměnné v jazyce PHP**

V každém programovacím jazyce je potřeba uchovávat hodnoty a dále s nimi pracovat. K tomu jsou využívány konstanty a proměnné, což je v podstatě pojmenování určité oblasti v paměti, kde je příslušná hodnota uložena. [2]

Konstanta je zástupný symbol pro nějakou hodnotu, na kterou je v kódu uveden odkaz a která je definována předem. Při pojmenování konstanty je nutné dbát na to, že musí začínat písmenem nebo podtržítkem a nesmí začínat číslem. U konstant je nutno rozlišovat mezi malými a velkými písmeny – názvy konstant jsou obvykle psány velkými písmeny. Hodnotu přiřazenou ke konstantě lze definovat pomocí funkce *define()*. Hodnotu jednou definované konstanty již nelze změnit ani ji zrušit. [1]

Na rozdíl od konstant se u proměnných počítá s tím, že se jejich hodnoty budou během zpracování skriptu měnit. Proměnné v PHP není třeba definovat ani deklarovat, stačí jim na vhodném místě přiřadit hodnotu. Fungují jako kontejnery uchovávající informaci pro pozdější použití ve skriptech, přičemž jejich obsah lze měnit. [1]

Proměnné jsou označeny symbolem dolaru (\$) a v jejich názvech se rozlišují velká a malá písmena. Za znakem dolaru musí následovat písmeno nebo podtržítko. [1] O přiřazení datového typu se stará samotné PHP. Datový typ proměnné se dokonce může v průběhu programu měnit. [2]

Za speciální typ proměnné můžeme označit pole, které umožňuje pod jedním názvem ukládat celou řadu hodnot. Prostřednictvím indexu, zadávaného do hranatých závorek pak máme přístup k jednotlivým hodnotám uloženým v poli. [1]

### 2.5.9 Funkce v jazyce PHP

V PHP existují jak funkce vlastní, které definuje sám uživatel, tak i funkce interní. Mezi základní interní funkce např. patří funkce *phpinfo()*, která vypíše řadu diagnostických údajů o instalaci PHP i daném prostředí. Podobně jako proměnná má i funkce svůj identifikátor, podle kterého je možné ji volat. Za identifikátorem následují vždy závorky, uvnitř kterých může být funkci předán parametr (nebo více parametrů). Funkce také mohou vracet hodnotu, kterou může být přímo výsledek (např. početní operace) nebo hodnota určující úspěch či neúspěch provádění funkce. Návratovou hodnotou nemusí být pouze celé číslo, může to být i reálné číslo, řetězec nebo objekt. Typ návratové hodnoty není třeba specifikovat, opět se o to stará samotné PHP. [2]

Funkci lze deklarovat na libovolném místě v programu, ovšem bývá zvykem uvést definici všech funkcí na začátku. Deklarace funkce má následující syntaxi:

```
function jmeno(seznam_parametrů) {  
    příkaz1;  
    příkaz2;  
    ...  
    return $hodnota;  
}
```

Pokud v těle funkce není nalezen příkaz `return`, je funkce ukončena dosažením složené závorky ukončující tělo funkce. V takovém případě funkce nevrací žádnou hodnotu. [2]

### 2.5.10 Podmínky v jazyce PHP

V rámci programu není možné používat pouze sledy příkazů a volání funkcí. Obvykle nastává potřeba provádět určité části programu pouze při splnění jistých podmínek nebo opakovat provádění některých úseků podle potřeby (viz cykly). [2]

Nejjednodušší příkaz pro větvení představuje příkaz `if`. Za tímto příkazem následuje podmínka, určující kdy bude následný příkaz nebo blok příkazů prováděn. Pokud je tento příkaz pravdivý, je provedena příslušná operace (nebo blok operací). Příkaz `if` lze také rozšířit o volitelnou část `else`, popřípadě `elseif`, které specifikují, co dělat v případě, že podmínka neplatí, resp. v případě, že platí jiná varianta. Syntaxe jednoduchého větvení pomocí podmínky `if - else` může vypadat takto:

```
if (podmínka1)
    příkaz;
elseif (podmínka2)
    příkaz;
elseif (podmínka3)
    příkaz;
else
    příkaz;
```

Samozřejmě místo jednotlivého příkazu může po podmínce následovat celý blok prováděných příkazů, ohraničený složenými závorkami. [2]

Speciálním typem větvení je rozcestník, *switch*. Ten snadno rozlišuje mezi mnoha variantami, které ovšem nejsou popsány pomocí podmínek, ale právě konkrétních hodnot. Na základě každé hodnoty pomocí klíčového slova *case* provádíme jeden či více příkazů, které mají být vykonány v zadaném případě. [2]

### 2.5.11 Cykly v jazyce PHP

K dalším možnostem řízení programu patří cykly. Jejich smyslem je umožnit opakovat určitou část programu. V PHP jsou k dispozici tři typy cyklů: *for*, *while* a *do-while*. [2]

Cyklus *for* provádí vždy zadaný počet opakování. Jemu podobným příkazem je cyklus *foreach*, který slouží k iterativnímu procházení všech prvků daného pole. [2]

Cyklus *while* se používá v případech, kdy není znám počet průchodů jako v případě cyklu *for*, ale pouze podmínka pro ukončení. [2]

Cyklus *do-while* je používán méně často, ale ve strukturovaném programování má své místo. Jedná se o neodmítavý cyklus, jeho tělo je vždy provedeno alespoň jednou. [2]

### 2.5.12 Soubory cookies v PHP

Jednou z vlastností protokolu HTTP je, že se jedná o protokol bezstavový. To znamená, že funguje v jednoduchém režimu typu požadavek-odpověď, aniž by jednotlivé požadavky na sebe byly nějak vázány. Pro běžné prohlížení stránek to není nijak omezující a v době návrhu WWW se také s mnoha možnostmi nepočítalo. Jedním z rozšíření, které se postupem času ujalo, jsou soubory *cookies*. Jedná se o krátké textové řetězce, které se mezi sebou vyměňují webový server a webový klient v rámci hlavičky HTTP. Server *cookie*

vytvoří a zašle ji klientovi, který ji u sebe uloží. Pokud klient příště přistupuje na stejný server, opět zašle tuto informaci v rámci požadavku HTTP serveru, jenž tak má možnost zjistit, že tento klient již na serveru byl. Zároveň je možné v těchto souborech ukládat nejružnější informace – počet přístupů, poslední prohlíženou stránku či identifikaci uživatele. Soubory *cookie* jsou uchovávány v běžných textových souborech na klientských počítačích a stejně tak jsou nezakódované přenášeny internetem. Nehodí se proto pro uchovávání citlivých údajů jako jsou hesla, čísla kreditních karet atd. *Cookies* lze klientovi odesílat pomocí funkce *SetCookie()*. Tato funkce má až sedm parametrů v následující syntaxi:

<code>SetCookie (jméno, hodnota, platnost, cesta, doména, bezpečnost);</code>
---

Jediným povinným parametrem je jméno. Obvykle se však používají ještě další dva parametry, hodnota a platnost. Vzhledem k tomu, že jsou *cookies* odesílány v hlavičce požadavku, je třeba funkci volat ještě před tím, než skript vypíše jakýkoliv text na výstup. [2]

### 2.5.13 Sessions v PHP

Podpora sessions je v jazyce PHP zakotvena až od verze PHP4, do té doby musela být nahrazována různými podpůrnými knihovnami. Základní myšlenkou sessions je, že k uživateli putuje pouze nějaký náhodný řetězec či velké číslo, které slouží k jeho identifikaci. Všechna data, která jsou spojena s prací uživatele na daném webu (např. počet návštěv, obsah košíku, uživatelské jméno) jsou uložena na samotném serveru, nejčastěji v databázi. Po zavření internetového prohlížeče obvykle data o dané session zanikají, stejně tak po delší době nečinnosti. Session může vzniknout automaticky návštěvou určitého serveru nebo teprve po přihlášení uživatele, to záleží na charakteru konkrétní aplikace. Identifikace session je obvykle velmi dlouhé jedinečné číslo – aby nemohlo dojít k záměně mezi dvěma uživateli. Pokud k serveru přistupuje uživatel již s platnou identifikací session, jsou vyhledána data týkající se dané session a následně obnovena do prostředí PHP tak, aby je aplikace měla k dispozici. Po dokončení skriptu jsou data opět automaticky uložena, aby byla k dispozici pro případnou další návštěvu téhož webu v rámci jedné session. Celý postup má řadu výhod oproti cookies, především pak:

- bezpečnost (citlivá data není potřeba přenášet mezi serverem a uživatelem),
- anonymitu (identifikace sessions je zcela anonymní),
- rychlost (není potřeba přenášet tolik dat v hlavičkách HTTP),



- spolehlivost.

Na všech stránkách, kde se plánuje sessions používat, je třeba (nejlépe úplně na začátku stránky) volat funkci *session\_start()*. Tato funkce zajistí právě to, že se pokusí od uživatele převzít identifikaci existující session a v případě úspěchu najde a zpřístupní data uložená pro danou session, zatímco v případě neúspěchu založí novou session a odešle uživateli její identifikaci. Proměnné, jejichž hodnoty se mají v rámci sessions uchovávat, je třeba předem určit. K tomu slouží funkce *session\_register()*. Jejím parametrem je jméno proměnné, kterou chceme do session zařadit. [2] Na proměnné session se odvolává pomocí syntaxe `$_SESSION[,proměnná]`. Pokus o odkazování na proměnnou přímým způsobem, bez použití této syntaxe by skončil neúspěchem – výsledkem by byla prázdná proměnná a mohla by být zobrazena případně také varovná zpráva. [1]

#### 2.5.14 Práce s databází v PHP

Jak již bylo výše uvedeno, PHP přímo podporuje databázový systém MySQL. Všechny funkce, které s tímto databázovým systémem pracují lze poznat snadno podle toho, že začínají písmeny „mysql“. [2]

Základními funkcemi pro práci s MySQL, které byly použity v projektu, jsou především tyto (včetně parametrů v závorkách):

- *mysql\_connect(server, uživatel, heslo)* – slouží pro připojení k serveru,
- *mysql\_select\_db(jméno\_databáze)* – výběr databáze,
- *mysql\_query(dotaz, připojení)* – pokládání dotazu,
- *mysql\_affected\_rows()* – počet řádků ovlivněných dotazem,
- *mysql\_num\_rows()* – počet vybraných řádků (u dotazu SELECT),
- *mysql\_fetch\_array()* – výběr jednoho řádku v asociativním poli,
- *mysql\_free\_result()* – uvolnění dotazu z paměti.[2]

## 2.6 Databáze

V dnešní době jsou nejčastěji používané tzv. relační databáze. Jejich výhodou je snadná ovladatelnost a dobrá reprezentace a podpora běžných datových struktur. Relační databázový systém umožňuje uživatelům vytvářet jednotlivé samostatné databáze, přičemž každá z nich může obsahovat jednu či více tabulek. Tabulky jsou tvořeny řádky a sloupci.

Řádky odpovídají jednotlivým záznamům, datovým větám, a sloupce atributům, jednotlivým prvkům těchto datových vět. Znamená to, že všechny datové věty v jedné tabulce mají stejnou strukturu atributů danou strukturou tabulky. [2]

### 2.6.1 Klíče

Klíč je údaj, v němž se každá hodnota může vyskytovat pouze jednou. Klíčem může být hodnota z jednoho sloupce, často se lze také setkat s klíči, jež jsou kombinací hodnota z několika sloupců. Klíč tedy jednoznačným způsobem označuje každý řádek v tabulce, protože žádné dva řádky nemohou mít stejný klíč. Každá tabulka může mít pouze jeden speciální klíč, který slouží jako primární jednoznačný identifikátor tabulky – primární klíč (primary key). Primární klíč nesmí obsahovat prázdné hodnoty (null) a musí být jedinečný. [2]

### 2.6.2 Relace a referenční integrita

Pokud mají dvě tabulky souviset, musí obě obsahovat sloupec, který je spojuje. Existují tři typy relací: 1:1 (jeden k jednomu), 1:N (jeden k mnoha), N:N (mnoho k mnoha). Relace 1:N je nejčastěji používanou relací, zatímco relace 1:1 se vyskytuje spíše ojediněle. Relace N:N je v podstatě spojením dvou relací typu 1:N prostřednictvím spojové tabulky. [1]

Je – li aplikace založena na referenční integritě, pak se záznamy v jedné tabulce odkazují na data v jiné tabulce. Druhá tabulka tedy obsahuje související záznam. Je – li z první tabulky odstraněn záznam nesoucí odkaz na záznam v druhé tabulce, je referenční integrita ztracena. [1]

### 2.6.3 Datové typy v mySQL

Databáze typu MySQL vychází z definice jazyka SQL (Structured Query Language, strukturovaný dotazovací jazyk), který je v podstatě standardem pro přístup k datům v relačních databázových systémech. V PHP je rozhodně nejpoužívanější databáze MySQL, v zásadě lze ale použít prakticky jakoukoliv databázi, PHP má prostředky pro komunikaci s velkým množstvím různorodých databázových systémů a navíc podporuje standard ODBC. [2]

K dispozici jsou základní druhy datových struktur – celočíselné datové typy (TINYINT, SMALLINT, MEDIUMINT, INT, BIGINT), čísla s pohyblivou řádovou čárkou (FLOAT, DOUBLE, DECIMAL), datové typy „Datum a čas“ (DATE, DATETIME,

TIMESTAMP, TIME, YEAR), řetězce (CHAR, VARCHAR), rozsáhlá datová pole (TINYBLOB, TINYTEXT, BLOB, TEXT, LONGBLOB, LONGTEXT), výčty (ENUM) a množiny (SET). K atributům lze kromě datových typů přiřadit také modifikátory – NOT NULL (atribut je vždy vyžadován, nelze vložit hodnotu NULL), UNIQUE (vynutí jedinečnost hodnoty v daném sloupci), PRIMARY KEY (primární klíč, musí být unikátní a nesmí chybět), AUTO\_INCREMENT (systém automaticky vygeneruje unikátní hodnotu v pořadí), DEFAULT (nastaví standardní hodnotu pro daný atribut v případě, že hodnota nebude zadána). [2]

#### 2.6.4 Práce se strukturou a daty v databázích

Nově vytvořená databáze je samozřejmě prázdná. Taková je sama o sobě k ničemu, proto je v ní třeba vytvořit tabulku (nebo více tabulek) pro ukládání dat. K tomuto účelu slouží v SQL příkaz CREATE TABLE se syntaxí:

```
CREATE TABLE tabulka (název_sloupce1 typ [modifikátory], název_sloupce2 typ [modifikátory], ...)
```

Během práce s databází může nastat situace, že jedna či více tabulek nemá požadovanou strukturu a je potřeba ji změnit. Nejčastěji bývá třeba přidat nový sloupec nebo změnit jméno či datový typ stávajícího sloupce. K úpravám slouží příkaz ALTER TABLE, lze pomocí něj přidávat, modifikovat i mazat sloupce tabulky.

```
ALTER TABLE tabulka ADD/CHANGE/DROP název_sloupce [modifikátory]
```

Mazání tabulky se provádí pomocí příkazu DROP TABLE se zadáním jména tabulky, která má být smazána. Obnova ze zálohy je obvykle možná, přesto by s tímto příkazem mělo být zacházeno opatrně. [2]

Předchozí příkazy slouží pro manipulaci se strukturou tabulky a databáze, následující se používají k manipulaci s obsahem tabulky. Pomocí příkazu INSERT se do tabulek vkládají nová data. Jeden příkaz INSERT umožňuje vkládat data pouze do jedné tabulky – pokud je potřeba vložit data do více tabulek, je nutné použít více příkazů. Základní syntaxí příkazu INSERT je:

```
INSERT INTO tabulka (seznam_sloupců) VALUES (seznam hodnot)
```

Někdy je naopak třeba z tabulky nepotřebné záznamy smazat. K tomu se v jazyce SQL používá příkaz `DELETE` s následující syntaxí:

```
DELETE FROM tabulka [WHERE podmínka]
```

Podmínka se většinou váže k nějakému atributu, nejčastěji to bývá identifikační číslo nějakého záznamu. Pokud by podmínka nebyla specifikována, budou smazány všechny záznamy tabulky. Lze použít i více podmínek zároveň. K jejich spojení se používá logických spojek `AND` (logický součin) a `OR` (logický součet), často ve spojení s běžnými operátory (např. `<`, `>`, `=`). Za speciální typ operátoru může být považován výraz `IS NULL` (ověřuje, zda je zadaná hodnota vyplněna), resp. analogicky `IS NOT NULL`. Při práci s textovými řetězci se často užívá operátor `LIKE`, který porovnává zadané znaky a umožňuje použít zástupné znaky „%“ a „\_“. [2]

Pro úpravu záznamů tabulky je určen příkaz `UPDATE`. Jeho pomocí lze zadaným záznamům změnit příslušné atributy na nové hodnoty. Syntaxe tohoto příkazu je:

```
UPDATE tabulka SET atribut=hodnota [WHERE podmínka]
```

Nejkomplexnějším příkazem jazyka SQL je ovšem výběrový příkaz `SELECT`. Zároveň ho můžeme považovat za nejpoužívanější příkaz ze všech, neboť se jedná a jediný příkaz, který umožňuje provádět výběr z databáze, čtení dat. Zatímco změna struktury dat tabulky se provádí jen zřídka, výběr dat je nejčastější operací. Syntaxe příkazu `SELECT` je následující:

```
SELECT seznam atributů  
FROM seznam tabulek  
[WHERE podmínky]  
[GROUP BY atribut]  
[HAVING podmínky]  
[ORDER BY atributy]  
[LIMIT [začátek], počet]
```

Jak je vidět, větší část příkazu `SELECT` je nepovinná (uvedeno v hranatých závorkách). V každém případě ale za příkazem `SELECT` následuje seznam atributů, které mají být použity ve výběru. Lze použít i `*`, což znamená vybrání všech atributů tabulky. Další povinnou částí je pak příkaz `FROM`, který určuje tabulku (nebo seznam propojených tabulek), jež jsou do dotazu zahrnuty. Podmínka `WHERE` se používá velmi často, mj. i pro spojování tabulek. Klausule `GROUP BY` má význam při sumarizování výstupu podle určitého argumentu, klauzule `HAVING` se pak používá pro specifikace podmínek v takovýchto dotazech. Na závěr dotazu se pak obvykle využívá klauzule `ORDER BY`, která určuje řazení dat podle daného atributu, standardně jsou data tříděna vzestupně (`ASC`), pomocí modifikátoru `DESC` lze data seřadit sestupně. Klausulí `LIMIT` je vybrán pouze omezený počet řádků, přičemž může být zadán i začátek omezení, např. `LIMIT 10,10` vybere 10 záznamů počínaje 11. řádkem výběru. [2]

### **3 Představení a rozbor současné verze**

V této části práce je nejdříve představen sportovní klub, pro který jsou webové stránky určeny, a následně přes historii tvorby těchto stránek popsána aktuální verze, její přednosti a nedostatky.

#### **3.1 Představení šachového klubu**

Ještě před začátkem popisu technických detailů, řešení a podobných odborných věcí je nutno podat pár základních informací o cílové organizaci, pro kterou vše vzniká. Čtenář tohoto textu si pak lépe udělá obrázek o některých nárocích a požadavcích, které jsou specifické pro webové prezentace organizací tohoto typu.

Jak již bylo zmíněno v úvodní části této práce, šachový klub TJ MSA Dolní Benešov byl založen několika nadšenci v roce 1947. Rozebírat historii, úspěchy a neúspěchy a podobné věci přesahuje rámec této práce, nicméně je na tomto místě vhodné zmínit aktuální situaci v klubu a principy jeho chodu a řízení. K 1. 1. 2012 je v klubu evidováno 47 aktivních a 14 neaktivních členů. Neaktivními členy se rozumí šachisté, kteří se neúčastní oficiálních soutěží ŠSČR. Šachový klub je součástí tělovýchovné jednoty společnosti MSA a.s., která se zabývá výrobou průmyslových armatur a čerpadel. Kromě šachistů patří do této organizační složky např. i stolní tenisté, volejbalisté a další sportovci. Počtem členů se šachový klub řadí na 1. místo mezi všemi těmito oddíly sdruženými pod hlavičkou TJ MSA. Řídící strukturou šachového klubu je sedmičlenný výkonný výbor, který je volen valnou hromadou klubu na funkční období 2 let. V čele výboru stojí předseda klubu, který je taktéž volen na valné hromadě. Členy výboru jsou dále místopředseda klubu, pokladník, správce šachového materiálu, správce webových stránek a další řadoví členové.

#### **3.2 Historie stránek šachového klubu**

Počátky tvorby webových stránek pro ŠK TJ MSA Dolní Benešov sahají někdy do přelomu let 2004 a 2005. Tehdy byl zadán žákům 1. ročníku průmyslové střední školy v Opavě, oboru informatika, mezi které patřil i autor této práce, úkol vytvořit jednoduché statické stránky, bez nutnosti umisťovat je na internet. Požadavky na tento projekt nebyly nijak vysoké, samozřejmě většina studentů byla začátečníky bez jakýchkoliv znalostí tvorby webu, čehož si byly zadavatelé dobře vědomi. Webové stránky měly být napsány v jazyce HTML, bez nutnosti používat kaskádové styly, validace kódu byla nepovinným parametrem zadání (a to rozhodně celé věci neprospělo). Důraz byl kladen z velké většiny na vzhled a

obsah stránek, nikoliv na způsob tvorby. Většina projektů byla tvořena pomocí rámců (frames) nebo pomocí tabulek (tables). Pomocí kaskádových stylů se tehdy odvážili tvořit svůj projekt pouze nejzkušenější jednotlivci, mnoho jich ale nebylo. Nebylo tomu tak ani u první verze stránek šachového klubu.

První verze stránek byla dokončena počátkem roku 2005. Základní charakteristikou bylo, že se jednalo o statické webové stránky, vytvořené za použití tabulkového layoutu v obyčejném poznámkovém bloku, které nebyly publikovány na webu. Úroveň odpovídala zkušenostem autora, jež byly na začátečnické úrovni. Nicméně i přesto byl tento prvotní projekt připravován s cílem dříve nebo později jej uvést do „ostrého“ provozu na webu. Vzhledem k tomu, že osud takového projektu nebyl předem jistý a jak rozpočet autora návrhu, studenta, tak rozpočet šachového klubu nemohl podobné „pokusy“ financovat, bylo rozhodnuto o umístění na neplacenou doménu. Jako velká výhoda se pak ukázala možnost využít nabídky, se kterou přišel poskytovatel internetového připojení v lokalitě bydliště autora – pro účely šachových stránek prakticky neomezený webový prostor na svém serveru, a to bezplatně! Jedinou nevýhodou byla poměrně těžce zapamatovatelná adresa: <http://sachy-msa.dolnibenesov.net>. Nicméně pro běžný provoz to překážka nebyla, server fungoval s relativně vysokou spolehlivostí a stránky běžely bez problémů.

Co se týká základní technických parametrů a konstrukcí, bylo použito řešení pomocí tzv. tabulkového layoutu. Tento způsob tvorby webových stránek byl kdysi populární pro svou jednoduchost a možnost přesného popisu všech prvků webové stránky, nicméně v době tvorby první verze šachových stránek už byly zavedeny jako standard pro tvorbu webu kaskádové styly. Základ každého jednotlivého dokumentu v jazyce HTML byla tabulka o dvou sloupcích o celkové šířce 875 pixelů na šachovnicovém pozadí stránky. Hlavičku tabulky tvořil jednoduchý obrázek v levém horním rohu, který zároveň sloužil jako odkaz na hlavní stránku, a stručný a výstižný nápis „ŠK TJ MSA Dolní Benešov“. Pod nápisem a logem byl vyhrazený prostor pro něco, co bychom mohli nazvat jednoduchým bannerem. Jednalo se o klasický HTML prvek, který v dnešní době nemá příliš časté využití (vzhledem k zastaralosti), tedy běžící text. Hlavní myšlenkou tohoto prvku bylo upozornit návštěvníky stránek na aktuální akce pořádané šachovým klubem, příp. informovat o důležitých akcích šachovém dění v blízkém okolí. Levý sloupec tabulky byl vyhrazen pro menu, tvořené odkazy na jednotlivé stránky webu. Menu nemělo žádnou logickou strukturu, jednalo se pouze o volně umístěné obrázky s hypertextovým odkazem. Pod menu následovalo logo šachového klubu. Pravý sloupec tabulky byl určen pro vlastní obsah stránky. Obvykle bylo potřeba tento

prostor ještě dále rozdělit na jednotlivé oblasti, proto se v tomto pravém sloupci téměř vždy vyskytovala další vnořená tabulka a v ní pak již konkrétní data.

**ŠK TJ MSA DOLNÍ BENEŠOV**

**PROPOZICE NA 5. TURNAJ KP ŽÁKŮ**

**MENU**

**Kalendář**

**Odkazy**

**Archiv**

**Družstva**

**Členové**

**Galerie**

**Kontakt**

**Šachový klub TJ MSA Dolní Benešov**

Uvítali byste zde "Návštěvní knihu"?  
 Ano, dobrý nápad! 65%  
 Ne, je to zbytečné! 35%  
 Celkem hlasovalo: 17  
 BlueBoard.cz

**Z** V neděli 5. 2. odehrála naše družstva celkem dva zápasy a to v okresním přeboru. "D" vyhrálo doma s **Hradcem nad Moravicí** přesvědčivě **6,5 - 1,5**, "E" bohužel podlehl v Opavě domácímu **Westu "B"** v poměru **5 - 3**. Zpravodaj zatím není k dispozici.  
 (napsal **Pavel Rakús**, 6.2.2006)

**Z** **Aktualizace výsledků družstev a kalendáře**  
 V pondělí 30. ledna proběhla po delší pauze opět aktualizace těchto stránek. Jsou doplněny kompletní výsledky všech **družstev**, přibýlo také mnoho nových událostí v **kalendáři**. Je zde i nová anketa, podle které se bude rozhodovat o případném vzniku návštěvní knihy.  
 (napsal **Pavel Rakús**, 30.1.2006)

**T** **Výsledky Memoriálu Karla Dihla a Lubomíra Bednářka**  
 V pátek 30. prosince se sešli přátelé z klubů MSA Dolní Benešov a Slezanu Opava, aby společně uctili památku dvou významných šachistů svých klubů. V rámci tohoto setkání se uskutečnil také 1. ročník tohoto memoriálu, kterého se zúčastnilo celkem 26 hráčů. Po domluvě se hrálo tempem 2 x 10 minut na 7 kol. Vítězem turnaje se stal s přehledem jeho největší favorit **Radek Caletka (2143)** se **6,5 body**, druhý skončil domácí **Pavel Rakús (2098)** se **6 body** a 3. poměrně překvapivě, ale zaslouženě **Pavel Žák (1935)** s **5 body**. Po turnaji následovalo přátelské posezení až do pozdních večerních hodin...  
 (napsal **Pavel Rakús**, 31.12.2005)

**T** **Vánoční turnaj v Malých Hošticích**  
 Naši šachisté zaznamenali 27.12.2005 na rapid turnaji v Malých Hošticích vynikající úspěch! Turnaje se zúčastnilo 34 hráčů převážně z okresu Opava a hrál se na 9 kol švýcarským systémem. Celkovým vítězem turnaje se stal **ing. Jan Vitásek (2107)** z **MSA D.Benešov**, který v celém turnaji ztratil jen půl bodu. Druhé místo obsadil překvapivě další hráč MSA **Jiří Klobouk (1816)** před Jiřím Bolackým ze Slavojce Poruba. Čtvrté až šesté místo patřilo rovněž šachistům MSA v pořadí Jiří Neděla, Michal Kurka a na 6. místě skončil **Miroslav Suchánek (1704)**, který byl se 6 body zároveň nejlepším mládežníkem v turnaji. Nejlepší hráč Slezanu Opava Jan Hromada skončil na 7. místě. Poháry pro 2. a 3. místo v kategorii mládeže si odnesli rovněž hráči z MSA Martin Dihel a Michal Vavřínek.  
 (napsal **Radomír Vavřínek**, 28.12.2005)

**Z** **5. kolo moravskoslezské ligy a krajské soutěže**  
 V neděli 18. 12. došlo k dalšímu mistrovskému střetnutí v 2. msl, kde se utkaly družstva **Frýdku - Místku "B"** a **MSA "A"**. Zápas skončil našim vítězstvím v poměru **5,5 - 2,5**. "B" vyhrálo v Opavě s místním **SSK "A"** také v poměru **5,5 - 2,5** a "C" bohužel podlehl doma **Westu Opava "A"** a to **5 - 3**. Hrál i "F", které zaznamenalo první vítězství, když porazilo **ŠK Bolatice "A"** vysoko **4 - 1**. Všechny výsledky a zpravodaje jsou k dispozici v sekci "Družstva"...[[družstva](#)]  
 (napsal **Pavel Rakús**, 31.12.2005)

**Z** **4. kolo okresního přeboru - obě družstva vyhrála!**  
 Tuto neděli se oběma družstvům dařilo - "D" porazilo venku **Slezan Opava "D"** a to poměrně vysokým rozdílem **6 - 2** a "E" si doma poradilo s **Kravařemi "A"** také vysoko **5,5 - 2,5**. Aktualizovaná stránka "Družstva" by mohla být k dispozici v pondělí nebo úterý...[[družstva](#)]  
 (napsal **Pavel Rakús**, 11.12.2005)

**Trénink (2 koncovky)**

**Diagram 1: Bílý na tahu vyhraje**

**Diagram 2: Černý na tahu vyhraje**

**Počet návštěv 17 16**

Obrázek 3.1 – První verze internetových stránek ŠK

Později se na stránkách objevily i různé předprogramované objekty, jako počítadlo návštěv a také různé ankety. Pokud by měly být popsány hlavní nedostatky této prvotní verze,



musely by být uvedeny výtky jak ke grafickému návrhu, tak k samotnému řešení v jazyce HTML, především pak nedostatky jako:

- obrovské obrázky menu, zabírající mnoho cenného místa pro další objekty,
- špatná hlavička tabulky postrádající logo,
- ne příliš vydařené grafické zpracování obrázků,
- velmi špatná úroveň provedení v jazyce HTML,
- používání již zastaralých elementů.

V roce 2006 vstoupil šachový klub do společného projektu s polským partnerem z města Rydułtowy, který byl financován z operačního programu Evropské unie pro rozvoj regionů. To s sebou přineslo potřebu zviditelnění na webu, která byla požadována ze strany financovatele tohoto projektu. Především díky tomuto impulsu se začaly věci hýbat správným směrem a postupně začala vznikat nová verze stránek.

**ŠK TJ MSA  
DOLNÍ BENEŠOV**

**MENU**

Interreg -

Akce -

Soupisky -

Elo -

Kontakt -

Galerie -

Odkazy -

Archiv -

Guestbook -

**Středoškolská liga v šachu, okres Opava, 2006/07**

Č.	Název školy	1.	2.	3.	4.	5.	6.	Celkem
1.	SPŠ Stavební	----	2	2,5	4	2	1,5	<b>12</b>
2.	SŠPU Praskova "A"	3	----	4	5	3	4	<b>19</b>
3.	Slezské gymnázium	2,5	1	----	3	2	3	<b>11,5</b>
4.	SŠPU Praskova "B"	1	0	2	----	0	2	<b>5</b>
5.	Mendelovo gymnázium	3	2	3	5	----	2,5	<b>15,5</b>
6.	Obchodní akademie	3,5	1	2	3	2,5	----	<b>12</b>

**Konečné pořadí (po odehraných 5. kolech):**

Poř.	Název školy	Z	V	R	P	Body	Skóre
1.	SŠPU Praskova "A"	5	5	0	0	<b>15</b>	19
2.	Mendelovo gymnázium	5	3	1	1	<b>10</b>	15,5
3.	Obchodní akademie	5	2	1	2	<b>7</b>	12
4.	Slezské gymnázium	5	2	1	2	<b>7</b>	11,5
5.	SPŠ Stavební	5	1	1	3	<b>4</b>	12
6.	SŠPU Praskova "B"	5	0	0	4	<b>0</b>	5

**!! VAROVÁNÍ !!**  
Ke správnému zobrazení stránek je zapotřebí mít nainstalovaný **Macromedia Flash Player 8**.  
Stáhnout si ho můžete [bezplatně zde!](#)

**NEJBLIŽŠÍ AKCE**

**AKTUÁLNĚ**

**TGP**

**EUROREGION SILESIA**

Autorem tohoto projektu je **Pavel Rakús**, pavel.rakus@seznam.cz, ICQ: 306-416-090. Vytvořeno pro ŠK TJ MSA Dolní Benešov, 2006.

Obrázek 3.2 – druhá verze stránek ŠK

Na návrhu a realizaci této verze se nejvíce podepsalo používání grafického WYSIWYG editoru Macromedia Dreamweaver. Z předchozí verze zůstal tabulkový layout, nápis v horní části nahradil jednoduchý obrázek se zakomponovaným názvem klubu. Velkou změnu představovalo především hlavní menu, vytvořené pomocí prvků ve Flashi, které umožňoval Dreamweaver vytvářet. Na rozdíl od první verze byla stránka (tabulka) rozdělena do 3 sloupců – levý byl vyhrazen pro menu a ikonu počítačového příslušenství, prostřední, na jehož pozadí byl obrázek s vodoznakem klubu, byl vymezen pro vlastní obsah stránky, a pravý sloupec pak tvořily další malé vnořené tabulky s různými informacemi. Ve spodní části pak byl prostor pro informace o autorovi a kontakt na něj. Velká změna přišla v roce 2007, kdy bylo na stránkách poprvé využito technologie PHP ve spojení s databází MySQL. Tato změna ovšem ovlivnila pouze několik stránek – většina jich zůstala v klasickém HTML. Stejně jako předchozí verze i tato měla poměrně hrubé nedostatky, z těch nejzávažnějších by se dalo zmínit:

- používání Flashe pro tlačítka menu (nekompatibilní v některých prohlížečích),
- pouze omezené využívání databáze,
- špatně rozvržení obsahu na stránce.

Největším problémem a nedostatkem stránek bylo, že se pořád jednalo o statické HTML stránky, bez jakýchkoliv možností zásahu uživatelů do databáze. V praxi to znamenalo především neúměrnou zátěž na osobu administrátora, který nejen že musel vytvářet nové a nové stránky (jak se měnily požadavky ze strany klubu), ale zároveň i tvořit obsah, případně převádět obsah získaný od ostatních uživatelů do HTML podoby. Proto přišel nápad vytvořit novou, již třetí verzi, která s úpravami funguje dodnes.

### **3.3 Představení současné verze**

Práce na 3. verzi internetových stránek šachového klubu MSA Dolní Benešov započaly někdy začátkem roku 2008. Cíl projektu byl jasný – vytvořit jednoduché stránky na principu redakčního systému, pomocí kterého by oprávnění uživatelé tvořili vlastní obsah stránek. Server, na kterém webová prezentace běžela, již technologie PHP v kombinaci s MySQL umožňoval použít, takže přechod ze statických na dynamické webové stránky si nevyžádal žádnou změnu adresy. Nová verze webového informačního serveru šachového klubu byla spuštěna k 1. březnu 2008.

### 3.3.1 Řešení rozvržení obsahu na stránce

Stránky opět dostaly podobu tabulkového layoutu, i když tentokrát již mnohem propracovanějšího a zahrnujícího větší množství buněk s informacemi v postranních panelech. I přesto, že pro začátek návrhu byl opět použit program Macromedia Dreamweaver, bylo upuštěno od metody, která charakterizovala předchozí verzi stránek – menu využívajícího přednastavené flashové prvky.

informační server šachového klubu v Dolním Benešově Datum: 28. únor 2008 16:42 hod. - Čtvrtek

## ŠK TJ MSA DOLNÍ BENEŠOV

Administrace | Návod k administraci | Kde nás najdete? | O webu

### HLAVNÍ MENU

- úvodní stránka
- návštěvní kniha
- seznam členů
- družstva ŠK MSA
- kontaktní info
- galerie - fotky
- odkazy na www
- archiv výsledků
- kalendář akcí

### Družstva

- "A" - krajský přebor
- "B" - krajský přebor
- "C" - krajská soutěž
- "D" - krajská soutěž
- "E" - okresní přebor
- stránka družstev...

### Kontakt

předseda: [Radomír Vavřínek](#)  
místopředseda: [Lubomír Bělka](#)  
pokladník: [Blážena Krieblová](#)  
správce klubovny: [Matouš Gerns](#)  
správce webu: [Pavel Rakús](#)  
další informace...

### Doporučený prohlížeč

### Detail události

#### "A" vybojovalo cenné vítězství

Nedělní zápas byl opravdu zajímavý. Po tuhém boji se nakonec projevila naše elová převaha a houževnatého soupeře, který přijel v 7, jsme porazili 5 - 3...

vloženo: 25. 02. 2008 autor: Pavel Rakús

A zápas to rozhodně nebyl jednoduchý. Fryčovice sice přijeli v 7, což asi nejvíc vadilo naší největší posile, Zbyszkovi Wieczorkovi, který tentokrát na 1. desce vyhrál bez boje!  
Na 2. šachovnici hrál Jacek s černými proti Zdeňkovi Šnytovi a v partii celou dobu hájil mírně horší pozici - nakonec ji uhájil. Na třetí šachovnici jsem já bílými více než 60 tahů zkoušel Borovičku (ne tu borovičku, to bych asi nedal, ale toho Borovičku!), partii jsem ale nakonec stejně nevyhrál.  
Na 4. desce uputně bojoval Pavlík Kozák s Tomášem Lipovským, který má v této sezoně výborné výsledky. V zahájení se dostal do velmi špatné pozice a nikdo už mu nevěřil, že dokáže z partie něco vydolovat. S ubíhajícími tahy ale obětoval kvalitu, dostal soupeře pod tlak a nakonec **zvítězil** a zajistil tak v podstatě vítězství týmu.  
5. šachovnice - Valtr - v zahájení výborná, skoro vyhraná pozice, která vyústila v zisk pěšce, pak ale Valtr něco přehlédl a po ztrátě figury se koncovka už nedala moc dobře hrát.  
Na 6. šachovnici Jenda Vitásek opět špatně rozehrál zahájení a partie celkem rychle přešla do koncovky. V ní ztratil pěšce a nějakým nedopatřením později i figuru a bez odporu prohrál.  
Na 7. šachovnici předvedl náš kapitán Jarda Melecký prakticky bezchybný výkon a vyhrál, pravda proti slabšímu soupeři, ale to jsme koneckonců měli všichni...  
8. šachovnice - Luboš Bělka - s černými figurami se o jeho výkon člověk opravdu strachuje. Luboš se dostal do horší pozice, ale bitvu zvládnul takticky, soupeř mu odmítnul remízu a Luboš pěknou kombinací získal figuru a později vyhrál.

Po tomto tříbodovém vítězství 5 - 3 se tak družstvo definitivně zachránilo (když už teda ten postup nějak nevyšel) a pomohlo k záchraně také "B" družstvu. Příští zápas se hraje za 14 dní v Bruntále...

Pavel Rakús

### Nejbližší akce

#### 1. března 2008

KP družstev st. žáků (3. turnaj)  
V sobotu 1. března se v Havířově hraje další turnaj série krajského přeboru v rapid šachu družstev starších žáků. Bližší informace prozatím nejsou a účast Benešova není jistá.  
[Propozice](#)

### Aktuálně

Vítejte na zbrusu nových stránkách Šachového klubu TJ MSA Dolní Benešov!  
Nový jednoduchý redakční systém umožňuje vkládat články, propozice a výsledky. Zájemci si mohou požádat o už jméno a heslo na adrese správce webu!

### Narozeniny slaví...

04.03. - Tomáš Miketa - 18 let  
05.03. - Michal Bartoš - 48 let  
16.03. - Henryk Boldys - 15 let  
**Gratulujeme!**

### Návštěvní kniha

Poslední vzkaz byl napsán:  
2008-02-27 10:15:23 (admin)  
Celkový počet příspěvků: 1

### Náš turnaj

pořádají  
**ZÁVĚREČNÉ TURNAJE**  
**krajských přeborů**  
**do 10, 12, 14 a 18 let**  
**Přihlásit hráče do turnaje!!**

Autorem tohoto projektu je **Pavel Rakús**. Vytvořeno pro ŠK MSA D.B., 2008.

Obrázek 3.3 – třetí verze stránek ŠK

Barevné schéma stránek se inspirovalo stránkami šachového turnaje Corus Steel Chess. Bylo použito podobných barev a také velmi podobného rozložení, i když v případě stránek šachového klubu bylo opět využito metody vnořování tabulek. Záhloví stránek tentokrát doznalo významné změny – z pouhého obrázku s názvem klubu bylo změněno na textové, s plným názvem, vodorovným menu s odkazy na hlavní ovládací strukturu systému, zobrazením data a času a logem klubu v pravé části. Stránka, která byla opět rozdělena do

tabulky se třemi sloupci, z nichž dva krajní byly koncipovány jako postranní informační panely a třetí uprostřed sloužil pro zobrazení vlastního obsahu stránky. V levém sloupci se nacházely jednotlivé tabulky s požadovaným obsahem – hlavní menu, tvořené pomocí odrážek s odkazy na příslušné stránky, seznam družstev opět s odkazy na příslušnou stránku, dále kontakt na vybrané členy oddílu a reklama na doporučený prohlížeč (později nahrazena počítačem přístupů). Pravý postranní panel sloužil jako informativní, v tabulkách byly zobrazeny doplňující informace o nejbližší konané akci, text o aktuálním dění v klubu, tři členové, kteří brzy oslaví své narozeniny, kdo a kdy napsal poslední vzkaz do návštěvní knihy a grafika v podobě animace s upoutávkou na významnou akci, kterou šachový klub v blízké době pořádá. V zápatí byla informace o autorovi stránek a roku vytvoření projektu. Později byly některé prvky v obou postranních panelech na základě nových požadavků změněny. Doporučený prohlížeč byl zaměněn za počítačlo přístupů na stránky, jak již bylo zmíněno výše, a změn se dostalo také pravému sloupci, kde došlo k záměně pořadí některých prvků a náhradě aktuální informace za informace o společném projektu s polským partnerem v rámci přeshraniční spolupráce, financované z EU.

### **3.3.2 Formátování stránek**

Pro formátování textu, odkazů, formulářových prvků a tabulek bylo využito nově hlavně kaskádových stylů. Tuto možnost předchozí verze zcela ignorovala. Seznam vlastností kaskádových stylů byl uložen v externím souboru, ve zvláštních případech, kdy si to situace vyžadovala, bylo použito i přímého formátování na úrovni daného elementu stránky. Standardním fontem celého webu bylo zvoleno nejpoužívanější písmo Arial ve velikostech 10, 12 a 14 bodů. Rámečky tabulek byly upraveny pomocí vlastnosti *border-collapse* na šířku jednoho pixelu, hypertextovým odkazům bylo nastaveno odlišné formátování (tučné písmo a podtržení) při najetí myši nad daný odkaz. U seznamů bylo použito vlastních obrázků u odrážek, speciální podmíněné formátování bylo použito u tabulek s výsledky družstev. Zde již ovšem bylo využito jazyka PHP.

### **3.3.3 Administrační rozhraní**

Administrační rozhraní umožnilo po ověření uživatele vstup do chráněné části webových stránek a manipulaci s obsahem databáze. Ověření probíhalo pomocí dialogového okna HTTP Autentizace, kde uživatel zadal příslušné uživatelské jméno a heslo. Tyto údaje byly následně porovnány s údaji v databázi a při korektním vyplnění byla uživateli nabídnuta úvodní stránka s možnostmi administrace. O přidělení uživatelského jména a hesla bylo

potřeba zažádat u administrátora stránek, systém neumožňoval vytvoření vlastního účtu. Na hlavní stránce s administračními možnostmi se nacházela graficky odlišená tabulka s jednotlivými odkazy na formuláře pro manipulaci s databází a soubory.

### **3.3.4 Správa článků, akcí a archivu**

Hlavní pilíře informačního serveru šachového klubu tvořily články konkrétních autorů, členů klubu. Na úvodní stránce *index.php* byl zobrazen seznam deseti nejnovějších článků formou zkráceného výpisu z databáze s uvedením data zveřejnění článku, jménem autora, ilustračním obrázkem dané akce a odkazem na detail samotného článku, který představoval vlastní text a nepovinný odkaz na výsledky na webu. V administračním prostředí se po přihlášení uživatel dostal na svou základní stránku, kde podle nastavených práv mohl manipulovat s obsahem jednotlivých tabulek v databázi. Běžný uživatel měl k dispozici odkaz na seznam svých vlastních článků, které mohl následně editovat nebo případně smazat, a také odkaz na vytvoření nového článku. Správce webu a předseda oddílu pak měli práva nastaveny takovým způsobem, že jim bylo navíc povoleno editovat či mazat veškeré články, které byly v databázi uloženy.

Pro vkládání textu do databáze byl určen jednoduchý formulář. Data získaná z tohoto formuláře byla přeposlána na kontrolní stránku, kde byla kontrolována korektnost zadaných informací. V případě úspěšné kontroly byla data uložena do databáze a článek byl ihned k dispozici na úvodní stránce webu, v případě nevyplnění povinných údajů nebo nekorektním vyplnění formuláře byla data zaslána zpět na původní stránku pro zadávání a uživatel dostal možnost svou chybu opravit a pokusit se o odeslání znovu.

Velmi podobným způsobem fungovala i správa akcí a archivu. Pojmem správa akcí se rozumí formulář pro vkládání nových událostí do kalendáře akcí, který tvořil důležitou část webových stránek. Jedním z mála rozdílů byl odkaz na další pomocný formulář, který umožňoval k dané položce databáze (položka kalendáře nebo archivu) načíst také soubor, nejčastěji ve formátu *\*.doc* nebo *\*.pdf*, a do databáze zapsat odkaz na tento soubor. Správci měli opět možnost přístupu ke všem položkám v databázi.

### **3.3.5 Správa návštěvní knihy a družstev**

Tyto dvě sekce patřily k těm omezeným, přístup do nich měl pouze administrátor. Správa návštěvní knihy obsahovala stránku se seznamem příspěvků vložených do knihy během posledních 14 dní s výpisem data a času vložení, autora příspěvku a úryvkem z textu

příspěvku v délce 20 znaků. Vzhledem k tomu, že návštěvní kniha byla otevřená i všem návštěvníkům stránek a pro vkládání příspěvků do ní nebylo potřeba žádné registrace, mohly se v databázi sporadicky objevit i příspěvky s vulgárním nebo urážlivým podtextem, které bylo nutno co nejdříve smazat. Před smazáním příspěvku bylo nutno tuto akci potvrdit pomocí dialogového okna v JavaScriptu.

Správa družstev byla navržena jako jednoduchý formulář, ve kterém se zobrazil poslední odehraný zápas daného družstva. Pokud ještě nebyl odehrán, což bylo kontrolováno pomocí aktuálního data a data konání zápasu, nebylo možné vložit výsledek do databáze. V opačném případě správce zvolil z rozbalovacího seznamu počet získaných bodů a průběžné umístění, z čehož systém automaticky vyhodnotil zápas jako vyhraný, prohraný či remízový, spočítal aktuální počet bodů družstva a výsledek byl ihned k dispozici v tabulce na stránce daného družstva.

### **3.3.6 Seznam členů**

Další důležitou stránkou, která byla velmi důležitá pro informovanost návštěvníků, byl seznam členů. Jednalo se o stránku s výpisem všech členů klubu a jednoduchým formulářem, pomocí kterého bylo možné tento výpis filtrovat a řadit podle jednotlivých parametrů. Jméno člena zároveň fungovalo jako odkaz na detailní stránku s kartou člena, kde byly vypsány i bližší údaje o hráči jako adresa, telefon či e-mailová adresa.

### **3.3.7 Galerie fotografií**

Velký zájem samozřejmě byl i o možnost prohlédnout si aktuální fotografie z různých akcí a společných setkání. Fotogalerie byla koncipována jako jedna velmi rozsáhlá stránka s tabulkou v prostoru pro hlavní obsah stránky. V této tabulce byly vždy 3 řádky vyhrazeny pro jednu danou fotogalerii – v prvním byl název fotogalerie společně s odkazem na stránku, kde bylo možno fotky prohlížet, ve druhém místo a datum konání a ve třetím řádku pak tři náhodné miniatury fotek z dotyčné galerie. Po těchto třech řádcích následoval oddělovač v podobně vodorovné čáry a další 3 řádky jiné galerie.

### **3.3.8 Návštěvní kniha**

Stránka, která byla zavedena po konzultaci s vedením oddílu, si hned od počátku získala velkou oblibu. Obsahovala jednak jednoduchý formulář pro zadávání vzkazů a zároveň i seznam 20 posledních příspěvků, takže ihned po odeslání příspěvku do databáze byl tento zobrazen na dané stránce. Vzhledem k tomu, že někteří uživatelé do návštěvní knihy

opakovaně psali nevhodné vzkazy, které již bylo velmi obtížné rychle a účinně odstraňovat, byla zde přidána i možnost kontrolovat uživatele podle IP adresy a těm méně slušným odeprít možnost přidávat nové vzkazy. V měsíci dubnu roku 2012 počet příspěvků v návštěvní knize překročil číslo 3200.

### 3.3.9 Ostatní stránky

Další stránky samozřejmě měly svůj význam a důležitostí nemusely vůbec zaostávat za ostatními. Jednalo se ovšem především o statické stránky a většinou pouze s nepatrným využitím databází. Mezi tyto stránky můžeme zařadit *kontaktní info* (stránka obsahovala kontakty na řídící členy klubu), *ke stažení* (odkazy na soubory pro stažení, povětšinou formuláře), *odkazy na www* (seznam odkazů na jiné webové stránky) a *GENS UNA SUMUS* (informační stránka o česko-polském projektu stejnojmenného názvu z roku 2009). Horní menu nabízelo také odkaz *Kde nás najdete*, kde byla pomocí obrázků popsána cesta do hrací místnosti v Dolním Benešově a *O webu*, tam si návštěvníci mohli přečíst malé info o samotných webových stránkách.

### 3.3.10 Návrh databáze

Databáze byla navržena velmi jednoduchým způsobem. Spojení tabulek bylo využíváno pouze tam, kde to bylo potřeba – např. mezi tabulkou s uživatelskými účty a tabulkami se vkládanými daty. Velká část tabulek tak stála samostatně bez jakýchkoliv vazeb na další tabulky. V databázi bylo využíváno pro účely stránek šachového klubu 13 tabulek s celkovým počtem 4901 záznamů o velikosti 1.6 MB (k 23. 4. 2012).

## 3.4 Rozbor současné verze

V této části práce je popsán komentář k výše uvedenému představení aktuální verze stránek. Vzhledem k tomu, že se jedná o verzi, která má být nahrazena vylepšenou verzí, text v této kapitole je zaměřen na nedostatky aktuálních stránek. Neřeší už ovšem jejich odstranění, toto je předmětem a podstatou návrhu nové verze.

Nejdříve je potřeba v krátkém seznamu uvést jednoduchý obecný popis nedostatků, kterými současná verze trpí. Mohou mezi ně být zařazeny:

- problémy tabulkového layoutu,
- nevyřešený systém uživatelských účtů,

- zásadní chyby v galerii fotografií,
- problémy při kontrole zadaných údajů,
- problémy při využívání databáze,
- nedostatky u zobrazovaných seznamů,
- pouze omezená míra automatizace,
- nedostatky v kódování,
- grafické zpracování.

### **3.4.1 Problémy tabulkového layoutu**

Tabulkový layout byl kdysi hojně využíván, dnes se již od toho systému upouští. Důvodem je především to, že při využívání tabulkového layoutu je celá tabulka nejdříve načtena a zobrazena až právě po načtení. To může způsobit situaci, kdy uživatel čeká na stránku neúměrně dlouhou dobu. Layout současné verze stránek byl ovšem navrhnout poměrně nešťastně, což má za následek nestejně vysoké postranní panely a nestejně rozestupy mezi jednotlivými moduly (tabulkami) v těchto panelech. Výsledný vzhled tak působí mírně nepravidelným dojmem.

### **3.4.2 Problémy se systémem uživatelských účtů**

Toto je jeden z nejzávažnějších problémů současné verze. Údaje o uživateli jsou sice uloženy v databázi, ale vždy se o to musí postarat administrátor až na žádost (e-mail) daného uživatele. Heslo v databázi není žádným způsobem šifrováno, což je také nepřípustné. K uživatelským účtům nejsou nastavena žádná práva, jsou pouze rozlišovány dva konkrétní účty, které mají oprávnění správce, ostatní mají pouze běžné možnosti.

### **3.4.3 Chyby v galerii fotografií**

Stránka *galerie.php*, na které je umístěna tabulka s miniaturami fotek a odkazy na jednotlivé fotogalerie, vykazuje zásadní nedostatky. Velkým problémem se ukazuje být zobrazení všech galerií na jedné webové stránce, která se díky obrovskému počtu obrázků, byť miniaturních, velice dlouho načítá. Dalším nedostatkem je výběr miniatur do tabulky, který je generován náhodně. Chybí již ovšem zamezení ve výběru stejného obrázku dvakrát (či dokonce třikrát) za sebou, což působí nepřirozeným dojmem. Za nedostatek může být označena i značná redundance stránek se zobrazením fotek, protože ty se liší pouze v detailech, a není nutné pro každou vytvářet vlastní stránku.



#### **3.4.4 Nedostatky v kontrole zadaných údajů**

Kontrola zadaných údajů probíhá přes více stránek než je potřeba. V některých případech není kontrola dostatečná a uživatel má možnost zadat nepřípustné znaky do databáze. Formulář na stránce s návštěvní knihou obsahuje textové pole pro opsání textu z obrázku, ten je však neměnný a proto může být taková kontrola neúčinná.

#### **3.4.5 Problémy při využívání databáze**

Návrh databáze velmi málo využívá propojování více tabulek, což zvyšuje redundanci dat, protože ty pak musí být obsaženy na několika místech najednou. V databázi se některé tabulky vyskytují nadbytečně, údaje z nich by bylo možno sloučit do jedné tabulky za přidání dalšího atributu (sloupce) tabulky. Nedostatkem je také malý obsah databáze, který ovšem úzce souvisí i s dalšími problémy stránek.

#### **3.4.6 Nedostatky u seznamů**

Seznamy položek, získaných z konkrétních tabulek databáze, tvoří podstatnou část stránek jak v administračním prostředí, tak i ve veřejně přístupných částech webu. Zásadní problém pak představuje zobrazení všech položek na jedné stránce a také chybějící možnost seznamy filtrovat a řadit podle zadaných atributů.

#### **3.4.7 Omezená míra automatizace**

Potřeba automatizace stála za vznikem této verze stránek. Od roku 2008 došlo v této oblasti k některým důležitým změnám, které opět pomohly zlepšit míru automatizace a tím ulehčily administrátorovi život. Nicméně míra automatizace, která v tomto případě souvisí i s možností spravovat jiné uživatelské účty apod., jistě nedosahuje nejvyšší možné úrovně a lze ji dále vylepšovat.

#### **3.4.8 Nedostatky v kódování**

Bohužel stejně jako předchozí verze i ta současná trpí nedostatky v kódování. HTML kód není na všech stránkách validní, je zde výskyt některých starších prvků a implementace CSS není úplně stoprocentní. K nedostatkům v kódování je potřeba zařadit také někdy zbytečně dlouhé kódy skriptů jazyka PHP a jazyka SQL, které lze vyřešit kratším a přehlednějším zápisem.

### **3.4.9 Grafické zpracování**

Grafické zpracování některých stránek, především fotogalerie je nedotažené. Stránky působí jednoduchým a zastaralým dojmem. Množství grafických prvků neodpovídá moderně pojaté tvorbě webu. U administrace je použito kombinace příliš mnoha barev, což působí rušivým dojmem a snižuje to serióznost stránek.

## 4 Návrh řešení nové verze (PHP, MySQL) a jeho realizace

Práce na návrhu vylepšené verze webového portálu šachového klubu započaly v průběhu roku 2011. Celý projekt byl koncipován tak, aby bylo spuštění nové verze stránek možné provést nejpozději ke konci května 2012, resp. k začátku června. Nová verze prakticky vychází z verze staré, velkou změnou ovšem prošel vzhled a návrh databáze, který si vyžádala změna systému uživatelských účtů a přihlašování. O porovnání obou verzí pojednává další kapitola této práce.

### 4.1 Produkční plán webu

Před zahájením samotných vývojářských prací byl stanoven jednoduchý produkční plán, jehož cílem bylo stanovit základní strategické cíle webu a další podrobnosti jako účel webu, rozpočet atd.

#### 4.1.1 Název webu a adresa

Název webu je poměrně jasný a shoduje se s názvem organizace, o které informační server pojednává. Celé znění tedy je: „*Šachový klub TJ MSA Dolní Benešov*“. Tak to stojí v titulcích všech stránek a pod tímto názvem ho znají i internetové vyhledávače. Adresa šachový stránek zůstává s příchodem nové verze stejná. Původní záměr byl upustit od této výše zmíněné internetové adresy 3. řádu a přejít na klasickou placenou doménu 2. řádu. Takové řešení ale bylo výborem klubu nakonec zamítnuto, z finančních důvodů.

#### 4.1.2 Záměr a cíle nového webu

Vznik staré verze stránek byl již popsán v kapitole o historii. Z uvedeného textu vyplývá, že hlavním cílem předchozí verze bylo ulehčit administrátorovi správu stránek a zavést prostřednictvím jazyka PHP a databáze MySQL jednoduchý informační systém, ve kterém by se spojili lidé z vedení klubu i běžní autoři a návštěvníci, s cílem sdílet mezi sebou všechny dostupné a relevantní informace z šachového i mimošachového dění v klubu a okolí.

Modernizace byla plánována již dříve, nicméně dochází k ní až v současné době. Za základní cíle nových stránek lze považovat

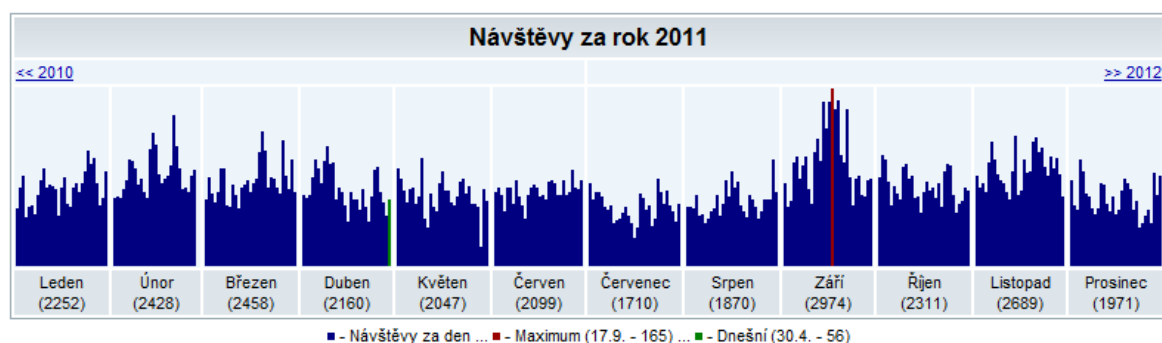
- postupné zvyšování návštěvnosti,
- přilákání většího počtu nových autorů,
- větší automatizaci základních procesů,

- větší zviditelnění klubu,
- získání nových sponzorů,
- umístění cílené placené reklamy.

Cílů by mělo být dosaženo především díky zlepšení funkčnosti systému a širším možnostem pro běžné návštěvníky. Kompletní změna designu také může pomoci některé cíle naplnit. Vzhledem k tomu, že celý projekt byl už od prvopočátku tvořen jako neziskový a pouze pro interní potřeby šachového klubu, je umístění cílené placené reklamy, byť v relativně nižším řádu než na hojně navštěvovaných serverech, spíše srdečným přáním autora (a možná několika dalších lidí z vedení klubu). Jiná situace ovšem nastává u získávání nových sponzorů. Zde je např. možnost vyhradit prostor pro zviditelnění případného sponzora na konkrétních stránkách webu nebo v případě větší podpory i na všech stránkách webového portálu.

#### 4.1.3 Cílové publikum a návštěvnost

Cílovými návštěvníky stránek jsou vzhledem k jejich zaměření především šachisté. Vzhledem k tomu, že se nejedná o tak populární sport jako např. fotbal, hokej či košíkovou, je možnost získávání nových uživatelů omezena. Pokud si navíc uvědomíme, že stránky slouží ve velké míře pouze pro informativní činnost uvnitř šachového klubu, pak jsou čísla o návštěvnosti poměrně slušná. Ve srovnání s konkurencí, za kterou by se daly považovat



**Obrázek 4.1 – graf návštěvnosti stránek ŠK**

webové stránky ostatních šachových oddílů v blízkém okolí (okres, kraj), je návštěvnost nadprůměrná. Z grafu návštěv za rok 2011 (viz obrázek 4.1) ovšem jasně vyplývá velký rozptýl počtu přístupů v jednotlivých obdobích roku. Je to způsobeno především dobou a délkou trvání šachové sezony a také tím, že přes letní měsíce se informace na webových stránkách mění jen velmi zřídka. Trend kolísání návštěv v průběhu roku také dokazuje zvýšený zájem návštěvníků v měsíci září, kdy šachový klub pořádá krajský přebor v rapid

šachu, na který je potřeba přihlásit se pomocí speciálního formuláře. Po rozšíření stránek o nové možnosti je nicméně očekáváno zvýšení návštěvnosti o několik desítek procent. Průměrná návštěvnost webových stránek činí v současnosti 66,96 jednotlivých uživatelských přístupů za den.

#### **4.1.4 Tým, konkurence, rozpočet**

Tvorba webového portálu šachového klubu TJ MSA Dolní Benešov zcela spočívá na bedrech autora této práce. Od vytvoření návrhu a realizace grafického řešení jednotlivých stránek, přes programování v jazyce PHP a tvorbu databázi v MySQL, až po ladění a vylepšování kódu i vzhledu na základě uživatelských reakcí a požadavků. To je práce poměrně náročná jak časově, tak psychicky. Podle stanov šachového klubu náleží za vykonávání práce správce webových stránek tomuto administrátorovi finanční odměna, vyplácená jednou ročně. Jedná se o odměnu spíše symbolickou, ale na běhu klubu se podílí spousta lidí a většinou na úkor svého osobního volna. Do týmu administrátorů může být zařazen i předseda šachového oddílu, jenž má přístup k některým funkcionalitám webu a stará se i o správu dat. Samozřejmě také v rámci svého volného času a bez nároku na odměnu.

Pokud vše funguje na principu „neziskovosti“, konkurenci ji těžké najít. Tedy alespoň konkurenci v pravém smyslu souboje dvou subjektů o přízeň zákazníka. Pokud bychom brali pojem konkurečního boje velmi obecně tak zjistíme, že podobných stránek informačního charakteru funguje v dnešní době již mnoho, a skoro všechny šachové oddíly v okolí se již na webu prezentují. Některé na lepší, některé na horší úrovni. Ovšem vždy za těmito projekty stojí pracovití lidé, kteří tuto činnost vykonávají dobrovolně a pouze za symbolickou či dokonce žádnou odměnu. Konkurence se proto projekt neobává.

Finanční rozpočet projektu není potřeba zmiňovat. Za návrh webu nebyla šachovému klubu účtována žádná sazba, za administraci pak náleží správci celoroční odměna, na jejíž výši nemá spuštění nové verze žádný vliv. Co se týká nákladů na provoz webu, ty pro klub nepředstavují žádnou zátěž, neboť doména a prostor je poskytován firmou p. Dušana Porwola zcela zdarma, za což mu patří velké díky.

## **4.2 Návrh grafického uspořádání stránek**

Navrhnout rozumné uspořádání webových stránek bývá většinou kámenem úrazu začínajících autorů. I když ani autor nemá s tvorbou designu stránek velké zkušenosti, bylo

rozhodnuto nevyužít žádnou ze šablon pro tvorbu webových stránek, kterých je dnes na internetu k dispozici celá řada, ale vytvořit stránky kompletně od začátku.

Předchozí verze, fungující od roku 2008, využívala tabulkový layout. To již bylo zmíněno, včetně výhod a nevýhod této metody. V současné verzi je již plně implementováno použití kaskádových stylů – nejen pro grafickou úpravu písma, tabulek, odkazů, ale i pro samotné rozložení stránky.

Stránka je tvořena třísloupcovým layoutem s pevnou šířkou sloupce. Layout je vytvořen pomocí plovoucího umístění. Princip spočívá v tom, že dva bloky, které jsou zobrazeny vpravo, jsou obaleny dalším elementem, který je umístěn taktéž vpravo a k němuž je přibalen další blok – levý postranní panel – opět pomocí plovoucího umístění (doleva). [4]



Obrázek 4.2 – layout nové verze stránek ŠK

Layout je zobrazen na obrázku 4.2. Jeho celková šířka je 1000 pixelů, přičemž pro levý postranní panel je vyhrazeno 160 pixelů, pro zobrazení vlastního obsahu webu je

používán prostřední blok o šířce 640 pixelů a zbývajících 200 pixelů připadá na pravý postranní panel. Stránku dále tvoří záhlaví o výšce 130 pixelů a zápatí o výšce 20 pixelů. V základní podobě má layout výšku 885 pixelů.

Rozložení obsahu by se dalo popsat spíše jako ortodoxní, založené na tradičním vnímání webové stránky uživatelem. V záhlaví je zobrazen název stránky a logo šachového klubu. Pro umístění jednotlivých elementů v záhlaví bylo využito absolutního pozicování prvků. Na obrázku je zobrazena pouze kostra stránek v kódování HTML s využitím CSS, chybí zde proto ještě blok s formulářovými prvky pro přihlášení uživatele a některé další prvky, které se objevují až ve verzi v PHP (zobrazení dne a data). V záhlaví lze najít jednoduché vodorovné menu, tvořené vždy kombinací obrázku s průhledným pozadím ve formátu PNG a odkazem na požadovanou stránku – administraci webu (tato část je přístupná pouze po ověření uživatele), informace o webu, popis cesty ke klubovně ŠK a jednoduchým formulářem pro vyhledávání na webových stránkách šachového klubu. Levý postranní panel slouží jako rozcestník stránek. Pro označení jednotlivých částí panelu (menu, družstva, návštěvy atd.) je opět využíváno obrázku PNG s průhledným pozadím, který je tvořen malým symbolem a stínovaným textem ve fontu *Dungeon*. V horní části panelu je zobrazeno hlavní menu s odkazy na další webové stránky. Každý odkaz je umístěn ve vlastním bloku `<div>` a tvoří ho obrázková odrážka a vlastní text odkazu, který po najetí myši změní formátování. Podobným stylem jsou tvořeny i odkazy na výsledky všech družstev šachového klubu. V levé části jsou dále ještě odkazy na kontaktní osoby šachového klubu a modul se zobrazením počítadla návštěv. Posledním prvkem je obrázek informující o validitě dané stránky. Prostřední blok je nositelem obsahu každé konkrétní stránky. V závislosti na požadavku uživatele jsou zde zobrazovány výpisy z databáze v podobě různých seznamů a tabulek, obrázky, odkazy na jiné webové portály, formuláře pro vkládání dat a další prvky. Pravý postranní panel má informativní charakter, odkazy se v něm objevují pouze zřídka. Nahoře je umístěna grafická reklama na nejbližší významnou akci, pořádanou šachovým klubem. Tento banner je tvořen za sebou uspořádanými obrázky GIF, které jsou v určitém intervalu obměňovány. Pod tímto objektem je většinou umístěn jeden až dva odkazy o stejném formátování jako odkazy hlavního menu, které slouží pro zobrazení propozic nebo stránky s výsledky či jiným materiálem z akce, kterou banner propagoval. Dále následuje blok s výpisem nejbližší šachové akce, vybrané z databáze. Pod tímto prvkem pak nalezneme krátký komentář k aktuálnímu dění, ať již přímo v šachovém klubu, nebo v jeho blízkém okolí. Další blok opět využívá napojení na databázi a zobrazuje návštěvníkům tři členy klubu,

kteří v blízké době oslaví své narozeniny. Poslední částí pravého panelu je pak element zobrazující výpis údajů o nejnovějším příspěvku, který byl vložen do návštěvní knihy, a také o celkovém počtu příspěvků, jež byly do knihy vloženy od doby jejího vzniku. V zápatí je zobrazen krátký text informačního charakteru, který upozorňuje na rok vzniku a autora tohoto projektu.

Je nutno podotknout, že se jedná o návrh, který bude uveden do praxe, a na základě diskuze s vedením klubu a dalšími relevantními osobami budou provedeny patřičné změny, které autor projektu uzná za přínosné a opodstatněné.

### **4.3 Kódování v HTML a validita**

Nové stránky jsou navrženy tak, aby každý jednotlivý dokument splňoval požadavky na kódování ve značkovacím jazyce HTML. Konkrétně podle předpisu HTML 4.01 Transitional. Od počátku psaní kódu byl kladen důraz na dodržování syntaxe a po dokončení dokumentu (a případném vložení skriptu PHP) byla provedena validace pomocí oficiálního validátoru, dostupného na webových stránkách konsorcia W3C. Díky ověření validity jsou nové stránky dobře čitelné i v případě vypnutí kaskádových stylů, jsou lépe čitelné pro roboty vyhledávačů jako Google nebo Seznam, zobrazují se korektně také na mobilních zařízeních a zaručují kompatibilitu i s budoucími verzemi internetových prohlížečů.

### **4.4 Návrh databáze MySQL**

Pro tvorbu databáze byl využit speciální programový systém phpMyAdmin, který je napsán v jazyce PHP a distribuován pod licencí GPL. Umožňuje vytvářet databáze a tabulky, manipulovat s nimi, provádět SQL příkazy a pracovat s klíči. [1] Před samotnou tvorbou kódu v PHP bylo zapotřebí navrhnout základní strukturu databáze v MySQL, do které budou patřičná data vkládána.

#### **4.4.1 Návrh struktury a datových typů tabulek**

Vzhledem k tomu, že cílem nového systému je jeho plná automatizace a přenechání ovládání uživatelům s příslušnými přístupovými právy, tvoří základ relační databáze tabulka s nutnými informacemi o registrovaných členech systému. Její struktura je popsána na obrázku 4.3. Sloupce jsou tvořeny potřebnými informacemi o uživateli, které je povinen při registraci do systému zadat. Všechny položky v tabulce jsou proto označeny jako NOT NULL, což znamená, že jejich vložení do tabulky je vždy vyžadováno – v opačném případě by



záznam nebyl vložen a byla by vyhlášena chyba. Identifikační číslo uživatele, jehož datovým typem je `INTEGER`, a které je zároveň primárním klíčem tabulky, je automaticky přiřazeno systémem na základě extra funkce `AUTO_INCREMENT`. Další sloupce tabulky jsou textového

	Sloupec	Typ	Vlastnosti	Nulový	Výchozí	Extra	Akce					
<input type="checkbox"/>	id	int(11)		Ne		auto_increment						
<input type="checkbox"/>	username	varchar(20)		Ne								
<input type="checkbox"/>	password	varchar(20)		Ne								
<input type="checkbox"/>	jmeno	varchar(20)		Ne								
<input type="checkbox"/>	prijmeni	varchar(30)		Ne								
<input type="checkbox"/>	email	varchar(40)		Ne								
<input type="checkbox"/>	obrazek	varchar(30)		Ne	default.jpg							
<input type="checkbox"/>	uroven	int(11)		Ne	3							
Zaškrtnout vše / Odškrtnout vše    Zaškrtnuté:												

Obrázek 4.3 – struktura tabulky `db_user`

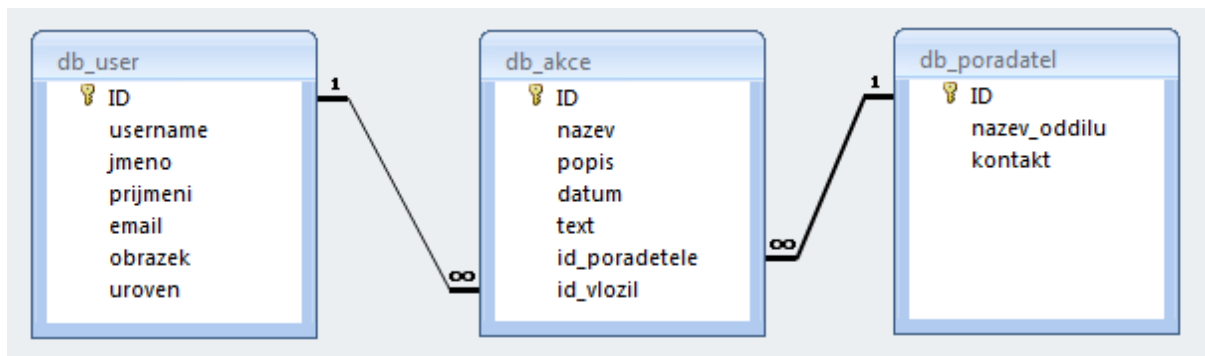
charakteru a slouží k uchování všech povinných údajů, které uživatel musí při registraci zadat. Jako prvek datové struktury byl zvolen datových typ `VARCHAR` s rozdílnou délkou vzhledem k tomu, že není a ani nemůže být dán žádný předpis pro velikost (rozuměno počet znaků) jednotlivých záznamů. Velikost je ovšem nastavena tak, aby nedošlo k nežádoucímu omezení uživatelů při výběru např. uživatelského jména nebo hesla. Na prvku `username`, do kterého je uloženo uživatelské přihlašovací jméno nebo přezdívkou, je nastaven index typu `UNIQUE`. Ten zajišťuje, že se v tabulce nevyskytnou dva záznamy o stejném `username`. Položky `obrazek` a `uroven` mají nastaveny výchozí hodnoty, které jsou při registraci nového uživatele uloženy v záznamu spolu s informacemi, které zadal pomocí registračního formuláře, a které byly zpracovány skriptem PHP. O jejich využití je zmínka dále.

Na podobném konceptu jsou založeny i ostatní tabulky, s jednou podstatnou výjimkou. Většina z nich, pokud je potřeba využití relace, obsahuje cizí klíč s vazbou na primární klíč jiné tabulky (nejčastěji je to samozřejmě tabulka se záznamy o uživateli). Kromě zmíněných datových typů jednotlivých sloupců jsou příležitostně využity i datové typy `DOUBLE` (pro desetinná čísla), `DATE` (jednoduché datum), `TIMESTAMP` (časové razítko), `ENUM` (výčet). Výhledově se uvažuje i o použití datového typu `BLOB` (binární dlouhý objekt) jako alternativní možnosti ukládání obrázků k uživatelským účtům.

#### 4.4.2 Vazby mezi tabulkami

Je potřeba si uvědomit, že databáze šachového oddílu se s počtem záznamů, který lze počítat v řádu jednotek tisíc, řadí k malým databázím, ve kterých je vyhledávání dílem

okamžiku. I přesto jsou na některých tabulkách vytvořeny indexy, které pomáhají lepší vyhledávání zajistit. Se spuštěním nové verze přihlašování je očekáván nárůst záznamů v databázi v důsledku toho, že dochází k rozšíření stránek, kde mohou uživatelé záznamy prostřednictvím formulářů vkládat. Zajištěním patřičných relací se snižuje duplicita záznamů v některých tabulkách, která byla typická pro předchozí verzi stránek. Relace používané v projektu jsou typu 1:N, kdy jeden záznam, který je primárním klíčem tabulky, je přiřazen více záznamům jiné tabulky, ve kterých je pak jako cizí klíč.



Obrázek 4.4 – příklad relací v databázi MySQL

Na obrázku 4.4 je zobrazen příklad používané relace mezi jednotlivými tabulkami. Důležitá je tabulka *db\_user*, jejíž struktura již byla popsána. Její ID je jako cizí klíč uloženo v tabulce *db\_akce*. To umožňuje administrátorovi databáze zjistit, kdo danou položku do databáze vložil a podle toho vykonat různé akce. Na tabulku *db\_akce* navazuje také tabulka *db\_poradatel*, která je příkladem jak zamezit zbytečné duplicitě záznamů. Zatímco v předchozí verzi byly položky *nazev\_oddilu* (předtím *poradatel*) a *kontakt* přímou součástí tabulky *db\_akce*, nyní jsou odděleny v samostatné tabulce *db\_poradatel*. To znamená, že nyní je v případě situace, kdy je poradatel u více akcí stejný, uloženo v tabulce *db\_akce* pouze jeho ID jako číslo a nikoliv celý název a kontakt ve formě delších textů.

#### 4.4.3 Zálohování databáze

Důležitou novinkou, kterou předchozí verze neřešila, může být zálohování databáze. Stránky v PHP za použití MySQL, které běží již od roku 2008, sice za čtyři roky svého provozu žádný problém se záznamy nebo s databázovým systémem obecně nezaznamenaly, ale i přesto by mělo v nové verzi být zavedeno určité schéma systematické zálohy alespoň některých podstatných částí (tabulek) databáze pro případ nečekané události. Zálohováním databáze je v tomto smyslu myšlen export v rozhraní phpMyAdmin do vhodné struktury souborů. Pro zálohování bylo rozhodnuto o využití exportu do souboru *\*.sql*, do kterého rozhraní phpMyAdmin vypíše strukturu dané tabulky (pomocí `CREATE TABLE`) a také

záznamy v podobě příkazu `INSERT`. V případě, že by byla tabulka poškozena nebo zcela ztracena v důsledku nějaké mimořádné události, může být po zprovoznění databáze z tohoto SQL souboru znova obnovena. O pravidelnosti záloh bude rozhodnuto na základě statistiky o nových záznamech v databázi.

## **4.5 Programové řešení v PHP**

Programové řešení celého systému šachového klubu bylo navrženo a vytvořeno v programovacím jazyce PHP. Nemá smysl zde uvádět celé stránky kódu a komentovat ho – to by pro nezasvěcené čtenáře v podstatě znamenalo ztrátu času. Proto se tato část zaměřuje pouze na jednotlivé části systému a spíše na slovní vysvětlení konkrétních problémů.

### **4.5.1 Kombinace HTML a PHP**

Vzhledem k tomu, že se jedná o webové stránky, které mají být zobrazeny uživateli v internetovém prohlížeči, je nutné, aby byl kód v jazyce PHP předeveden do podoby srozumitelné pro server. To již bylo zmíněno, není třeba se tím dále zabývat. Většina jednotlivých PHP stránek je na serveru uložena v kořenovém adresáři a má stejnou strukturu, která se liší pouze v obsahové části. Proto bylo využito rozdělení stránek do čtyř hlavních modulů, kterými jsou záhlaví, zápatí, levý panel a pravý panel. Tyto moduly jsou na každé stránce zobrazovány stejně a je žádoucí, aby nebyly přímou součástí HTML kódu – to by v případě náhlé změny či pouze drobné úpravy vyvolalo nutnost upravovat HTML kód na všech dosud vytvořených stránkách. Každý modul je proto uložen jako soubor *\*.inc.php* ve složce moduly a do kódu každé PHP stránky vložen pomocí funkce *include()*. Tento postup zajišťuje, že při změně odkazu v pravém panelu proběhne změna na všech stránkách, a bude zobrazena vždy stejně.

### **4.5.2 Registrace a přihlašování uživatelů do systému**

Předchozí verze stránek sice umožňovala přihlášení do systému a jednoduchou administraci, nicméně uživatelský účet, pomocí kterého se pak konkrétní uživatel, chcete-li autor, do systému přihlašoval, musel být nejdříve na základě jeho požadavku vytvořen přímo v rozhraní phpMyAdmin správcem stránek. Tento systém byl ne příliš pohodlný, ale přinášel i své výhody. Pokud se správce rozhodl, že daná osoba, která o vytvoření účtu žádá, se pro funkci autora a komentátora, případně dokonce správce, nehodí, požadavek na vytvoření byl zamítnut a uživatel přístup do systému nezískal. S novou verzí přichází podstatné změny.

V novém návrhu bylo upuštěno od využívání HTTP Autentizace, která má sice výhodu v relativní jednoduchosti, ovšem na úkor bezpečnosti celých stránek. Jako lepší možnost se jevílo zařídit si přihlašování vlastními silami, s využitím HTTP Sessions, do kterých jsou ukládány informace o aktuálním uživateli. Kód, který se stará o kontrolu přihlášení uživatele a předávání informací dalším stránkám, je umístěn v záhlaví všech stránek. Vzhledem k tomu, že je záhlaví zahrnuto do každé stránky pomocí funkce *include()*, je případná změna ihned platná ve všech PHP stránkách. Samotné přihlašování probíhá



Obrázek 4.5 – přihlašování do systému

pomocí formuláře, který je umístěn v pravém horním rohu každé stránky. Pokud uživatel korektně nevyplní login a heslo, případně zadá neexistující login nebo login s neplatným heslem, je zobrazeno varovné hlášení o neplatném přihlášení a uživatel má možnost pokusit se o korektní vstup do systému znovu (viz obrázek 4.5, bod 1). Pokud je zadáno uživatelské jméno a heslo, které se shoduje s údaji získanými z databáze (obr. 4.5, bod 2) je nastavena hodnota `$_SESSION['logged']` na 1 a do dalších proměnných sessions (*login\_jmeno*, *login\_opravneni*) jsou uloženy hodnoty získané z databáze z tabulky o uživateli. Následně je místo formuláře pro přihlášení zobrazen obsah sessions, což jsou položky login, jméno, příjmení, oprávnění a odkaz na obrázek (obr. 4.5, body 3 a 4). Verze pro přihlášeného a nepřihlášeného uživatele se liší také odkazem ve spodní části panelu určeného pro login. Nepřihlášenému návštěvníkovi je nabídnuta pomocí odkazu na stránku *register.php* možnost založit nový účet, kde po korektním vyplnění všech potřebných údajů a vložení těchto dat do databáze získá možnost okamžitého přihlášení do systému jako „běžný uživatel“. Naopak již přihlášený uživatel má možnost přemístit se na stránku *update\_account.php*, kde si může např. změnit přednastavený obrázek za vlastní nebo upravit některé svoje údaje. Systém uživatelských účtů funguje na třech základních úrovních oprávnění – administrátor, prémiový

uživatel a běžný uživatel. Tyto tři úrovně je třeba důsledně rozlišovat především v administračním prostředí, o němž pojednává další bod práce.

### 4.5.3 Administrační rozhraní

Do administračního rozhraní se můžou uživatelé pokusit vstoupit po načtení stránky *admin.php*, na kterou odkazuje link v levé horní části každé stránky (vedle ikony s náradím). V takovém případě skript zkontroluje obsah proměnné *logged*, uložené v aktuální relaci. Pokud není nastavena, resp. její hodnota je rozdílná od 1, stránka administrace uživateli nabídne místo možnosti podílet se na běhu systému pouze chybové hlášení, že není přihlášen, a že pro vstup do této zóny je to potřeba. Pokud již je uživatel přihlášen, nastupuje na scénu systém tří úrovní uživatelského oprávnění. Ten je nastaven poměrně jasnými pravidly a možnostmi:

```
switch ($_SESSION['login_opravneni']) {  
    case 3: bez práv na administraci  
  
    case 2: vkládání, úprava, mazání vlastního obsahu  
  
    case 1: správa uživatelů (úprava práv, rušení účtů)  
            přístup k veškerému obsahu databáze  
}
```

Ze zjednodušeného úryvku kódu je vidět, jakým způsobem stránka pracuje. Po ověření zda je uživatel skutečně přihlášen, je z aktuální relace získána hodnota proměnné *login\_opravneni*, podle níž přepínač rozhodne, která část programového kódu se bude provádět. Běžný uživatel nemá možnost se podílet na tvorbě zpráv, článků či akcí v kalendáři, ale může požádat administrátora o udělení statusu prémiového uživatele. Ten již má povoleno vkládat do databáze vlastní zprávy, články, položky v archivu (soubory s výsledky), položky do kalendáře akcí atd. Svůj vlastní obsah má také vždy zobrazen v administračním prostředí a je schopen jej filtrovat a řadit podle potřeby. Články, archiv a akce může libovolně upravovat nebo případně mazat. Nejvyšší úroveň oprávnění, v případě těchto stránek označenou číslem 1, mají pouze administrátoři webu. Po vložení prvního účtu do databáze, kterým byl účet autora projektu a správce celého webu, musela být hodnota oprávnění manuálně přenastavena v prostředí phpMyAdmin na 1. Tím pádem byl vytvořen první „superuživatel“, který má k dispozici veškerá práva na celý systém. V seznamech jsou mu zobrazovány všechny

požadované položky všech uživatelů, které má samozřejmě možnost filtrovat, ať již právě podle jednotlivých uživatelů nebo např. podle typu události (o tom ještě dále), a které si může také libovolně seřadit. Má také možnost upravovat oprávnění ostatních uživatelů na základě jejich požadavků. Systém odesílání požadavků doposud nebyl rozpracován, nicméně návrh počítá buď s upozorněním administrátora prostřednictvím odeslání automatického e-mailu s předmětem žádosti a obsahem ve formě údajů o žadateli, anebo s vytvořením další tabulky v databázi, která by požadavky uživatelů shromažďovala a následně je zobrazovala administrátorovi v jeho administračním prostředí (zde bude rozhodnuto, až bude jasné, která možnost je vzhledem k počtu nových uživatelů úspornější). Kromě možnosti upravovat uživatele je k dispozici také mazání uživatelských účtů. To probíhá samozřejmě po potvrzení dialogového okna, vytvořeného v JavaScriptu. Administrátor dále může spravovat a libovolně upravovat jednotlivé odehrané zápasy družstev, zadávat výsledky těchto zápasů do databáze, spravovat návštěvní knihu, spravovat seznam členů, spravovat komentáře k článkům, spravovat text s aktualitou a další. V případě návštěvní knihy má možnost zařadit do databáze IP adresu návštěvníka, který následně bude zbaven možnosti psát další příspěvky. Další možnosti administrace budou samozřejmě přibývat s rostoucími požadavky a nároky ze strany uživatelů webu.

#### **4.5.4 Redakční systém**

Řešení redakčního systému navázalo na již fungující redakční systém z předchozí verze. Na základě uživatelských požadavků se nicméně rozrostla sada přednastavených obrázků pro zobrazení loga dané akce a novinkou je také zavedení možnosti nahrání vlastního obrázku. To probíhá přes speciální formulář v administračním prostředí konkrétního uživatele. Pokud soubor vyhovuje všem nastaveným podmínkám skriptu (velikost souboru, velikost obrázku v pixelech, koncovka souboru v grafickém formátu) je následně pomocí funkce *move\_uploaded\_file()* nahrán do složky */pict/clanky* pod vzorem *username\_ + jmeno\_souboru* a informace o něm jsou uloženy do databáze, společně s identifikačním číslem uživatele. Když se pak uživatel rozhodne napsat článek, jako ikonu si může zvolit právě nahraný obrázek. To přináší samozřejmě také větší atraktivitu a rozmanitost stránek, neboť místo 5-10 neustále se opakujících ikon článků se jich najednou na webu může objevit velké množství a ke každé dané akci ta nejlépe vhodná (např. u reportáže ze šachového turnaje se může objevit fotka vítěze apod.). Dalším novým prvkem v redakčním systému je možnost přidávat komentáře k článkům a zprávám. Tuto možnost mají pouze registrovaní uživatelé bez rozlišení úrovně oprávnění, narozdíl např. od návštěvní knihy, kde byla možnost

přidávat příspěvky ponechána všem návštěvníkům webu bez nutnosti registrovat se. V redakčním systému je také nově zaveden lepší systém kontroly korektního vyplnění informací. Kontrolována je délka názvu jednotlivých polí a zadávání některých položek přestalo být povinné, v databázi byla nastavena hodnota `NOT NULL` na *false*.

#### 4.5.5 Práce s databází

Velmi podstatnou část samotného PHP kódu tvoří práce s databází MySQL. V podstatě je to základní stavební prvek tohoto systému. Pro práci s databází je na každé stránce hned po zahájení relace zahrnut pomocí funkce *include()* také soubor *db.inc.php* ze složky *moduly*, ve kterém jsou funkcí *define()* definovány konstanty, potřebné pro vytvoření správného spojení s databází, např. `MYSQL_HOST`, `MYSQL_USER` atd. Tyto konstanty jsou pak využívány pro připojení k databázi prostřednictvím funkcí PHP pro práci s databází MySQL (začínají *mysql\_*). Připojení k databázi pomocí funkce *mysql\_connect()* je vytvářeno na každé stránce, protože databázi využívají jako zdroj informací také záhlaví a pravý postranní panel. Po načtení dat do proměnných, k čemuž je využívána konstrukce cyklu *while* ve spojení s funkcemi *mysql\_fetch\_assoc()* nebo *mysql\_fetch\_row()*, je výsledek pomocí funkce *mysql\_free\_result()* odstraněn z paměti. Na konci celého PHP skriptu je připojení k databázi ukončeno funkcí *mysql\_close()*, kde parametrem je aktivní připojení.

Pokud se jedná o vkládání nestandardních údajů je toto kontrolováno pomocí funkce *mysql\_real\_escape\_string()* při pokládání dotazu prostřednictvím *mysql\_query()*.

Práce s databází pak probíhá prostřednictvím samotných dotazů SQL. Zřejmě nejčastěji používaným je dotaz `SELECT`. Ten je v různých modifikacích využíván na každé stránce, pro každý seznam. Příkazy `INSERT INTO`, `UPDATE` a `DELETE` jsou pak používány pouze v administrační části, případně na stránkách, kde je vkládání do databáze pomocí příkazu `INSERT INTO` povoleno běžným uživatelům, nebo i všem návštěvníkům (komentáře k článkům a návštěvní kniha).

Další funkcí, která je nově zavedena, a která také plně využívá databázi, je funkce vyhledávání na webu. Formulář pro vyhledávání je na každé stránce a odkazuje na skript *search.php*. Ten po přijetí a vyhodnocení zadaných dat zobrazí buď chybovou hlášku podle druhu nedostatku, nebo zobrazí konkrétní položky, které se shodují v určitých parametrech se zadaným výrazem (u hledání v seznamu členů je to jméno a příjmení, u článků a položek v archivu pak název).

## **5 Zhodnocení řešení a porovnání**

Tato kapitola je věnována popisu obecného zhodnocení řešení, a také porovnání obou verzí stránek šachového klubu ve dvou základních bodech.

### **5.1 Zhodnocení řešení**

Nejdříve by mělo být zhodnoceno řešení nového projektu, jenž právě spatřuje světlo světa. Podařilo se vytvořit systém, který výrazně zlepšil popularitu i návštěvnost webového portálu šachového klubu, a díky možnosti vytvářet nové účty, bez nutnosti obtěžování kteréhokoliv administrátora, zpřístupní základní funkce systému všem uživatelům, kteří o to budou stát.

Samozřejmě technické řešení projektu není a ani nemůže být vždy stoprocentní. Pokaždé se najde někdo, kdo by danou problematiku řešil jinak, možná lépe, možná hůře, podstatná je funkčnost celého systému a spokojenost uživatelů.

Otázkou pro někoho může být také vzhled. Grafické řešení je vždy velice náročné, protože zalíbit se s ním všem bývá takřka nemožné. Zde je prostor pro diskuzi, případné podnětné návrhy mohou být a budou do budoucí vylepšené verze zapracovány.

Pokud vznese někdo z kompetentních uživatelů námitku k řešení kterékoliv funkce systému, je zde podobně jako u designu prostor k diskuzi. Návrhy pro zvýšení počtu funkcí budou dozajista konzultovány na zasedání výboru šachového klubu a ty podnětné budou následně zapracovány do systému při jeho první větší aktualizaci. Taková by mohla proběhnout někdy po skončení testovacího období.

### **5.2 Porovnání se starší verzí**

Porovnání se starší verzí je provedeno ve dvou různých samostatných celcích. Nejdříve v porovnání grafiky a designu a poté ve funkčním řešení obou verzí.

#### **5.2.1 Grafické porovnání**

Starší verze webu trpěla základními grafickými nedostatky, které nová verze napravuje. Tabulkové rozvržení obsahu stránky bylo nahrazeno praktičtějším řešením pomocí kaskádových stylů a plovoucího umístění prvků v layoutu. V daleko větší míře se začalo používat také obrázků pro označení jednotlivých modulů a také v záhlaví stránky. Nově byla zavedena i ikona webu, takže pravidelní uživatelé používající např. prohlížeč Mozilla uvidí i



logo stránek v záložkách svého oblíbeného prohlížeče. Velké změny se dostalo také samotnému grafickému rozložení některých stránek. Největší změna je patrná ve fotogalerii, která je nyní nejen tvořena zcela jiným způsobem programově, ale také doznala velkých grafických změn. V neposlední řadě povolení vlastních obrázků u článků a také rozšíření přednastavených obrázků pomůže oživit graficky novou verzi oproti staré.

### **5.2.2 Porovnání funkčnosti**

Pokud se budeme bavit o funkčnosti a nových možnostech zaváděné verze, jsou samozřejmě postaveny na verzi starší, ale výrazné chyby v kódu byly opraveny a některé vlastnosti byly přebudovány zcela od začátku. Jedná se především o systém přihlašování uživatelů, kde HTTP Autentizaci nahradilo používání HTTP Sessions, uživatelských relací, dále velký nárůst počtu tabulek v databázi, který souvisí jednak právě se zaváděním systému větší automatizace chodu stránek a dále rozšíření možností uživatelů vkládat např. obrázky ke svým účtům nebo přidávat ikony pro své články. Velkých funkčních změn doznala administrace – správci mají nyní možnost kontrolovat všechny uživatele, udělovat jim přístupová práva, nově také možnost přidávat do databáze členy šachového klubu a měnit jejich osobní údaje, přidávat text s aktualitou přímo do postranního panelu všech stránek. Byla zlepšena kontrola údajů, vkládaných do databáze. Ta byla také upravena, ve větší míře se začalo používat propojení různých tabulek za účelem omezení duplicity dat. Zapomenout nesmíme ani na řešení v HTML, které bylo konečně po letech uvedeno do korektní podoby za použití validátoru organizace W3C.

## 6 Závěr

Člověk je od přírody tvor líný, to je všeobecně známé. Takové pravidlo platí zvláště u lidí, kteří jsou zvyklí něco vynalézat a svou tvorbou si práci zjednodušovat. A nejinak tomu bylo při začátku vytváření tohoto projektu.

Projekt začal vznikat někdy začátkem roku 2012 a postupně se přes úvodní nezdary začal propracovávat do cílové podoby. Od návrhu rozložení obsahu webové stránky, přes složitý návrh databáze až po řešení přihlašování uživatelů a propojení jednotlivých stránek. Zda bylo řešení úspěšné, ukáže až ostrá zkouška a kritika, případně pochvaly, které se snesou na hlavu autora. Cílem nového webového portálu je přilákat větší počet uživatelů a autorů, větší počet návštěvníků, kteří budou denně hledat nové zajímavé komentáře, články a postřehy z dění uvnitř i vně šachového klubu a jeho okolí, a sami se také budou podílet na jejich rozvoji. To je důvod, proč nová verze vznikala, a pokud se tento cíl podaří naplnit, bude to pro autora projektu hřejivý pocit z dobře vykonané práce.

Cílem této práce pak bylo analyzovat a popsat současnou verzi systému a připravit návrh řešení nové verze, který by pak mohl být realizován a přinést nové možnosti a příležitosti. Rozbor současně fungující verze se vytvořit podařilo bez zjevných známek pochybení, analýza se opírala především o výčet nedostatků, což ovšem vyplývalo z logiky věci, protože pokud má být něco nahrazeno lepší verzí, není potřeba vyzdvihovat přednosti nahrazovaného předmětu. Řešení problematiky programování v PHP a MySQL je v okruhu laiků na tuto problematiku asi ne příliš čtivé, nicméně zvolený slovní popis je srozumitelnější než různé dlouhé ukázky kódu, na kterých by se sice podstata nově vytvářeného systému jistě dala popsat, ale rozsah práce by tím byl značně negativně ovlivněn a přitom by nebyla zajištěna vyšší kvalita práce. Hlavním cílem byla samozřejmě realizace popsaného návrhu a jeho uvedení do praxe. I tento těžký úkol se podařilo splnit a pouze čas ukáže, zda byla racionalizace systému úspěšná, nebo zda bude potřeba vést směr webového portálu jiným směrem.

Do budoucna se počítá se zapracováním návrhů kompetentních uživatelů a zlepšení fungování systému na základě ostrého testování v praxi a následných úprav, ať již drobnějších nebo úprav většího charakteru.

## Seznam použité literatury

### Knižní zdroje

- [1] BORONCZYK, T., E.NARAMORE, J. GARNER, Y. LE SCOUARNEC, J. STOLZ a M. K. GLASS. *PHP6, MySQL, Apache: Vytváříme webové aplikace*. Přeložil Bodgan KISZKA. Brno: Computer Press, 2011. ISBN 978-80-251-2767-4.
- [2] BRÁZA, Jiří. *PHP5: Začínáme programovat*. Praha: Grada Publishing, 2005. ISBN 80-247-1146-X.
- [3] CEDERHOLM, Dan. *Webdesign s webovými standardy*. Přeložil Jaroslav BLAŽEK. Brno: Zoner Press, 2004. ISBN 80-86815-15-3.
- [4] DOMES, Martin. *333 tipů a triků pro CSS*. Brno: Computer Press, 2009. ISBN 978-80-251-2360-7.
- [5] HLAVENKA, Jiří. *Vytváříme WWW stránky a spravujeme moderní web site*. 7. vyd. Brno: Computer Press, 2005. ISBN 80-251-0801-5.
- [6] KOSEK, Jiří. *HTML - tvorba dokonalých www stránek - podrobný průvodce*. Praha: Grada Publishing, 1998. ISBN 80-7169-608-0.
- [7] WHITE, Ron. *Jak fungují počítače*. Přeložil Aleš SAPÁK. Praha: SoftPress, 2003. ISBN 80-86497-48-8.

### Internetové zdroje

- [8] ZAJÍC, Petr. *PHP(1) – Historie a budoucnost*. Linuxsoft.cz  
[online] 27. 5. 2004, [cit. 2. 4. 2012]  
Dostupný z WWW: [http://www.linuxsoft.cz/article.php?id\\_article=171](http://www.linuxsoft.cz/article.php?id_article=171)
- [9] ZAJÍC, Petr. *PHP(2) – Jak to funguje*. Linuxsoft.cz  
[online] 28. 5. 2004, [cit. 3. 4. 2012]  
Dostupný z WWW: [http://www.linuxsoft.cz/article.php?id\\_article=172](http://www.linuxsoft.cz/article.php?id_article=172)

## Seznam zkratk

ASC	Ascending, řazení vzestupně
ASCII	American Standard Code for Information Interchange, americký standardní kód pro výměnu informací
BLOB	Binary Large Object, označení binárních dat v databázi
CERN	Conseil Européen pour la Recherche, evropská organizace pro atomový výzkum
CSS	Cascade Style Sheet, kaskádové styly
DESC	Descending, řazení sestupně
ENUM	Enumeration, výčet
EU	Evropská Unie
FI	Form Interpreter, překladač PHP
GIF	Graphics Interchange Format, grafický formát souboru
HTML	HyperText Markup Language, hypertextový značkovací jazyk
HTTP	HyperText Transfer Protocol, internetový protokol
HTTPS	HyperText Transfer Protocol Secure, zabezpečená verze HTTP
MB	MegaByte, jednotka kapacity informace
MSA	MoravskoSlezská Armaturka
ODBC	Open DataBase Connectivity, otevřená konektivita databáze
PHP	Personal Home Page, skriptovací jazyk
PNG	Portable Network Graphics, grafický formát souboru
SQL	Structured Query Language, strukturovaný dotazovací jazyk
SSL	Secure Sockets Layer, vrstva zabezpečených socketů, protokol
ŠK	Šachový Klub
ŠSČR	Šachový Svaz České Republiky
TCP	Transmission Control Protocol, protokol
TJ	Tělovýchovná jednota
TLS	Transport Layer Security, protokol

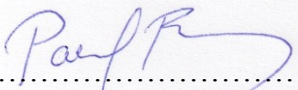
URL	Uniform Resource Locator, jednotný lokátor zdrojů
WWW	World Wide Web, celosvětová síť počítačů
WYSIWYG	What You See Is What You Get, typ editorů WWW stránek
W3	viz WWW
W3C	World Wide Web Consortium, mezinárodní konsorcium
XHTML	eXtensible HyperText Markup Language, rozšířená verze HTML
XML	eXtensible Markup Language, rozšiřitelný značkový jazyk

## **Prohlášení o využití výsledků bakalářské práce**

Prohlašuji, že

- jsem byl seznámen s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. – autorský zákon, zejména § 35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a § 60 – školní dílo;
- beru na vědomí, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen VŠB-TUO) má právo nevýdělečně, ke své vnitřní potřebě, bakalářskou práci užít (§ 35 odst. 3);
- souhlasím s tím, že bakalářská práce bude v elektronické podobě archivována v Ústřední knihovně VŠB-TUO a jeden výtisk bude uložen u vedoucího bakalářské práce. Souhlasím s tím, že bibliografické údaje o bakalářské práci budou zveřejněny v informačním systému VŠB-TUO;
- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu § 12 odst. 4 autorského zákona;
- bylo sjednáno, že užít své dílo, bakalářskou práci, nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).

V Ostravě dne 11. 5. 2012



Pavel Rakús

## **Seznam příloh**

Příloha č. 1:           Obrazová příloha

Příloha č. 2:           Zdrojové soubory na CD

## Příloha č. 1 – Obrazová příloha

# ŠK TJ MSA Dolní Benešov

Login:   
Heslo:

Administrace
O webu
Kde nás najdete
Hledat
založit nový účet

Hlavní menu

- úvodní stránka
- kalendář akcí
- seznam členů
- archiv výsledků
- fotogalerie
- kontaktní info
- odkazy na webu
- ke stažení
- návštěvní kniha

Družstva

"A" - 2. liga  
"B" - krajský přebor  
"C" - krajská soutěž  
"D" - okresní přebor  
"E" - okresní soutěž

Kontakt

předseda:  
Radomír Vavřínek  
správce webu:  
Pavel Rakús

### VIDEŇSKÁ SIMULTÁNKA S GM TOPALOVEM.

Během tří měsíců, třetí produkce-Markoš, Klíma, Topalov! Kdo bude další na řadě? [více...](#)

Melecký Jaroslav vloženo: 26.4.2012 počet komentářů: 6

### Tadeáš Kriebel vyhlášen nejúspěšnějším sportovcem roku!

Okresní svaz ČSTV Frýdek Místek vyhlásil nejúspěšnější sportovce roku 2011. [více...](#)

Melecký Jaroslav vloženo: 25.4.2012 počet komentářů: 1

### FINÁLE KP MLÁDEŽE V RAPID ŠACHU

V sobotu 31.3. přivítal velký sál kulturního domu v Dolním Benešově nejlepší krajské mládežnické šachisty a šachistky do 18 let, aby se utkali za šachovnicemi o tituly krajských přeborníků ve čtyř věkových kategoriích. [více...](#)

Vavřínek Radomír vloženo: 5.4.2012 počet komentářů: 6

### V Bohuslavicích vítězí GM Robert Cvek

S přehledem, samozřejmě! [více...](#)

Rakús Pavel vloženo: 25.3.2012 počet komentářů: 0

### BOHUSLAVICE OPEN 2012

Náš turnaj

**a Kulturní dům D.B.**

Výsledky na webu

Nejbližší akce

**11.5.2012**  
Ukončení sezóny ŠK TJ MSA

V pátek od 18.00 hod. se uskuteční v klubovně ŠK TJ MSA Dolní Benešov ukončení sezóny 2011-12. Po valné hromadě šachového klubu bude následovat volná zábava.

Aktuálně

**Aktualita**  
Aktuálně se nekoná žádná velká akce.

Narozeniny slaví

10.5. - Marcel Mirvald, 39 let  
18.5. - Josef Kostřica, 79 let  
18.5. - Jaroslav Žák, 62 let

# ŠK TJ MSA Dolní Benešov

Login:   
Heslo:

Administrace
O webu
Kde nás najdete
Hledat
založit nový účet

Hlavní menu

- úvodní stránka
- kalendář akcí
- seznam členů
- archiv výsledků
- fotogalerie
- kontaktní info
- odkazy na webu
- ke stažení
- návštěvní kniha

Družstva

"A" - 2. liga  
"B" - krajský přebor  
"C" - krajská soutěž  
"D" - okresní přebor  
"E" - okresní soutěž

Kontakt

předseda:  
Radomír Vavřínek  
správce webu:  
Pavel Rakús

### VIDEŇSKÁ SIMULTÁNKA S GM TOPALOVEM.

Během tří měsíců, třetí produkce-Markoš, Klíma, Topalov! Kdo bude další na řadě?

Melecký Jaroslav vloženo: 26.4.2012 počet komentářů: 6

V pátek 27. 4. se ve Vídní koná seminář k šachům do škol. Jde o celoevropský projekt k zavedení šachů jako nepovinného předmětu do základních škol. K propagaci akce se uskuteční simultánka, ve které bude Českou republiku reprezentovat FM Tadeáš Kriebel, který spolu s dalšími sedmi vybranými juniory z Evropy zasedne proti exmistru světa z roku 2005 velmistru Veselinu Topalovovi. Produkce se uskuteční v reprezentačních prostorách muzea Albertinum v centru města. Na partii mají hráči 90 minut s přidáváním času 30 sekund za každý vykonaný tah. Začíná se ve 12.45 a partie budou přenášeny on-line. Po skončení simultánky čeká její účastníky veřejný rozbor partií a závěrečná slavnostní večeře. Přímý přenos partií: <http://live.chess.at/> Webové stránky organizátora: <http://www.schachkurse.at/>

Odkaz na [www: http://live.chess.at/](http://live.chess.at/)

**Komentáře (6):**

Nejste přihlášen. Nemůžete vkládat komentáře. Přihlašte se v pravém horním rohu stránky, nebo si vytvořte nový účet [zde](#).

Paulie napsal 5.5.2012 v 02:23:  
Pořád to funguje !

Pavel Rakús napsal 5.5.2012 v 02:17:  
Konečně!!!

Náš turnaj

**a Kulturní dům D.B.**

Výsledky na webu

Nejbližší akce

**11.5.2012**  
Ukončení sezóny ŠK TJ MSA

V pátek od 18.00 hod. se uskuteční v klubovně ŠK TJ MSA Dolní Benešov ukončení sezóny 2011-12. Po valné hromadě šachového klubu bude následovat volná zábava.

Aktuálně

**Aktualita**  
Aktuálně se nekoná žádná velká akce.

Narozeniny slaví

10.5. - Marcel Mirvald, 39 let  
18.5. - Josef Kostřica, 79 let  
18.5. - Jaroslav Žák, 62 let



		<h1>Dolní Benešov</h1>		Login: <input type="text"/> Heslo: <input type="password"/> OK													
<a href="#">Administrace</a> <a href="#">Hlavní menu</a> <ul style="list-style-type: none"> <li>úvodní stránka</li> <li>kalendář akcí</li> <li>seznam členů</li> <li>archiv výsledků</li> <li>fotogalerie</li> <li>kontaktní info</li> <li>odkazy na webu</li> <li>ke stažení</li> <li>návštěvní kniha</li> </ul>	<a href="#">O webu</a> <a href="#">Družstva</a> <p>"A" - 2. liga            "B" - krajský přebor            "C" - krajská soutěž            "D" - okresní přebor            "E" - okresní soutěž</p>	<a href="#">Kde nás najdete</a> <div style="border: 1px solid black; padding: 5px;">  Registrace nového uživatele           <table border="1"> <tr> <td>Uživatelské jméno:</td> <td><input type="text"/></td> </tr> <tr> <td>Heslo:</td> <td><input type="password"/></td> </tr> <tr> <td>E-mail:</td> <td><input type="text"/></td> </tr> <tr> <td>Jméno:</td> <td><input type="text"/></td> </tr> <tr> <td>Příjmení:</td> <td><input type="text"/></td> </tr> <tr> <td>Alias:</td> <td><input type="text"/></td> </tr> <tr> <td colspan="2"><input type="button" value="Registrovat"/></td> </tr> </table> </div>	Uživatelské jméno:	<input type="text"/>	Heslo:	<input type="password"/>	E-mail:	<input type="text"/>	Jméno:	<input type="text"/>	Příjmení:	<input type="text"/>	Alias:	<input type="text"/>	<input type="button" value="Registrovat"/>		<a href="#">Hledat</a> <input type="text"/> <a href="#">založit nový účet</a>
Uživatelské jméno:	<input type="text"/>																
Heslo:	<input type="password"/>																
E-mail:	<input type="text"/>																
Jméno:	<input type="text"/>																
Příjmení:	<input type="text"/>																
Alias:	<input type="text"/>																
<input type="button" value="Registrovat"/>																	
		<b>Náš turnaj</b> 															
		<b>Výsledky na webu</b> 															
		<b>Nejbližší akce</b> <div> <b>11.5.2012</b>              Ukončení sezóny ŠK TJ MSA              V pátek od 18.00 hod. se uskuteční v klubovně ŠK TJ MSA Dolní Benešov ukončení sezóny 2011-12. Po valné hromadě šachového klubu bude následovat volná zábava.           </div>															
		<b>Aktuálně</b> <div> <b>Aktualita</b>              Aktuálně se nekoná žádná velká akce.           </div>															
		<b>Narozeniny slaví</b> <div>             10.5. - Marcel Mirvald, 39 let              18.5. - Josef Kostřica, 79 let              18.5. - Jaroslav Žák, 62 let           </div>															
<b>Kontakt</b> <p>předseda: Radomír Vavřínek</p> <p>správce webu: Pavel Rakus</p>																	

